

**БИБЛИОТЕЧКА
ПРОГРАММИСТА**

Д. УОЛШ

**Руководство
по созданию
документации
для математического
обеспечения**



БИБЛИОТЕЧКА
ПРОГРАММИСТА

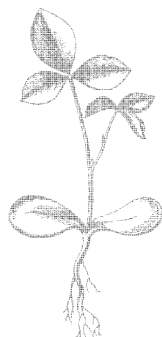
Д. УОЛШ

РУКОВОДСТВО
ПО СОЗДАНИЮ
ДОКУМЕНТАЦИИ
ДЛЯ МАТЕМАТИЧЕСКОГО
ОБЕСПЕЧЕНИЯ

Перевод с английского
под редакцией
Н. И. КОЗЛОВА



ИЗДАТЕЛЬСТВО «НАУКА»
ГЛАВНАЯ РЕДАКЦИЯ
ФИЗИКО-МАТЕМАТИЧЕСКОЙ ЛИТЕРАТУРЫ
МОСКВА 1975



Scan AAW

517
У63
УДК 519.95

Руководство по созданию документации для математического обеспечения, Д. Уолш, перевод с английского. Серия «Библиотечка программиста». Главная редакция физико-математической литературы изд-ва «Наука», 1975.

Основное содержание книги — модели текстовой документации, показывающие, как следует описывать те или иные элементы математического обеспечения ЭВМ, как составлять общие описания, описания операционной системы, описания библиотечных программ.

Книга предназначена для специалистов по системному программированию и математическому обеспечению, разработчиков машин, а также для студентов соответствующих специальностей.

Д. Уолш

**РУКОВОДСТВО ПО СОЗДАНИЮ ДОКУМЕНТАЦИИ
ДЛЯ МАТЕМАТИЧЕСКОГО ОБЕСПЕЧЕНИЯ**

(Серия: «Библиотечка программиста»)
М., 1975 г., 128 стр.

Редактор *Г. Я. Пирогова*
Техн. редактор *Л. В. Лихачева* Корректор *Е. Я. Строева*

Сдано в набор 11/III 1974 г. Подписано к печати 9/1 1975 г. Бумага 84×108¹/₃₂, тип. № 2. Физ. печ. л. 4. Условн. печ. л. 6,72. Уч.-изд. л. 6,37. Тираж 30000 экз. Цена 46 коп. Заказ № 113.

Издательство «Наука»
Главная редакция физико-математической литературы
117071, Москва, В-71, Ленинский проспект, 15

Ордена Трудового Красного Знамени Ленинградская типография № 2 имени Евгении Соколовой Союзполиграфпрома при Государственном комитете Совета Министров СССР по делам издательств, полиграфии и книжной торговли 189052, Ленинград, Л-52. Измайловский пр., 29

У $\frac{20204-024}{053 (02)-75}$ 72-74

© Перевод на русский язык,
Главная редакция физико-математической литературы
издательства «Наука», 1975.

ОГЛАВЛЕНИЕ

Предисловие редактора перевода	5
Предисловие автора	8
Как пользоваться этой книгой	10
Введение	11
Общие разделы	17
Модели документации	20
Модель 1. Подготовка описания технического задания на проект системы/программы	20
Модель 2. Подготовка руководства пользователю по работе с недиалоговой системой/программой	28
Модель 3. Подготовка руководства по работе с недиалоговой прикладной системой/программой	34
Модель 4. Подготовка описания внутренней логики	41
Модель 5. Подготовка руководства пользователю по диалоговой прикладной программе	46
Модель 6. Подготовка руководства пользователю по подпрограмме	56
Модель 7. Подготовка описания библиотеки программ	61
Модель 8. Подготовка описания языка ассемблера	64
Модель 9. Подготовка руководства пользователю по транслятору	73
Модель 10. Подготовка описания языка ввода-вывода	80
Модель 11. Подготовка руководства пользователю по операционной системе	88
Модель 12. Подготовка руководства оператору по операционной системе	105
Модель 13. Подготовка описания программного обеспечения	118
Модель 14. Подготовка проспектов	124
Приложение. Как писать руководство, используя модель документа	127

ТАБЛИЦА
КАТЕГОРИИ ПОЛЬЗОВАТЕЛЕЙ
ДОКУМЕНТАЦИИ ПО ПРОГРАММИРОВАНИЮ

Тип пользователя	Объяснение
Руководство по техническому программированию	Лицо, управляющее деятельностью по техническому программированию в Вычислительном центре
Профессиональный прикладной программист	Не требует пояснений
Профессиональный системный программист	Человек, который пишет системные программы или отвечает за них
Обучающийся программист	Человек, чья основная работа не включает программирование на ЭВМ, но тот, который должен время от времени писать программы
Профессиональный оператор ЭВМ	Не требует пояснений
Временный оператор	Человек, который должен при определенных обстоятельствах работать на ЭВМ, но не имеет опыта как оператор
Рабочий персонал	Лицо, обслуживающее установку. Планировщик машинного времени. Персонал, обслуживающий устройства записи. Персонал управления данными
Прикладной инженер	Опытный представитель заказчика
Торговый агент по ЭВМ	Не требует пояснений
Руководитель по нетехническому программированию	Не требует пояснений
Канторский персонал	Перфораторщик. Библиотекарь. Машинистка и т. д.

ПРЕДИСЛОВИЕ РЕДАКТОРА ПЕРЕВОДА

Характерной особенностью современного этапа развития вычислительной техники (ВТ) следует считать широкое внедрение ее методов во все сферы научной и производственной деятельности. Если на заре своего развития электронные вычислительные машины (ЭВМ) использовались в основном для проведения численных расчетов сложных инженерно-технических и научных задач, то сейчас в основном парк ЭВМ используется для экономических расчетов, для управления. Системы, работающие в реальном времени, с выносными пультами, с применением самых разнообразных внешних устройств, упрощающих взаимодействие человека с машиной, составляют основную массу вычислительных мощностей как за рубежом, так и в нашей стране.

Круг пользователей ЭВМ расширяется с огромной скоростью. Поэтому вопросы повышения производительности труда в программировании начинают играть не только научную, но и экономическую роль. ВТ становится новой отраслью производства.

На первом этапе использования ЭВМ основные трудности состояли в технической «слабости» ЭВМ. С развитием вычислительной техники центр тяжести в ней переместился в программирование. В связи с этим стали развиваться методы автоматизации программирования. Появилось мощное «программное обеспечение» (ПО), без которого эксплуатация современных ЭВМ немыслима. Стоимость этого «мягкого железа» поднялась от 3—7% стоимости всей вычислительной системы (ВС) до 50% и выше.

Развитие ПО имело далеко идущие следствия, одним из которых явилось серьезное расширение круга пользователей ЭВМ. Например, проблемно-ориентированные языки упростили задачу программирования и тем самым привлекли в ряды программистов новые силы за счет специалистов других областей.

Тем не менее вопрос о повышении производительности труда программистов сегодня не только не снят, а пожалуй, стоит даже более остро, чем раньше, поскольку теперь приходится изучать большое количество документов, описывающих ПО. Помимо хорошего знания, например, проблемно-ориентированного языка, для создания эффективной программы необходимо еще знать некоторые тонкости работы транслятора, так что необходимый объем знаний современного программиста вырос чрезвычайно сильно. Если к этому прибавить непрерывное обновление средств ПО, то становится очевидным, что появилась новая проблема «человек — ПО».

Программист сегодня сталкивается с документацией по ПО, совершенно не ориентированной на пользователя. Как правило, это научные отчеты о проделанной работе. На практике из всех возможностей, заложенных в ПО, не используется и половина, тратится огромное время на изучение ненужного материала, а опытных специалистов постоянно отрывают на консультации. Можно с уверенностью сказать, что сегодня дальнейшее совершенствование средств ПО упирается в создание разумной документации.

По этим причинам книга Дороти Уолш представляет большой интерес как с научной, так и с практической точки зрения.

Дороти Уолш, вице-президент АСТ (Advanced Computer Techniques Corporation), занималась вопросами, связанными с ВТ, с 1955 г. В 1960—1967 гг. она читала лекции по вычислительным методам и статистике в ряде учебных заведений США. Дороти Уолш проводила семинары по анализу программ, руководству в разработке проектов и стандартизации программирования для сотрудников фирмы АСТ и вела обучение системных программистов фирмы IBM по вопросам разработки и оформления документации.

Она провела год в одном из европейских филиалов АСТ, руководя организацией, разработкой и написанием на французском и английском языках руководств по ПО. В книге излагается весьма прогрессивный взгляд на подготовку документации. Интересна и нова для нашего читателя сама идея создания моделей документов. При быстром обновлении ПО

единый стиль создаваемой документации с помощью моделей позволит пользователю с меньшими усилиями освоить новые средства ПО. Идея моделей тесно связана с задачами стандартизации и унификации, столь необходимыми в ВТ и ее применениях.

Чрезвычайно важна постоянно подчеркиваемая мысль о необходимости ориентации создаваемых документов на задачи пользователей разных категорий. Неоднократно подчеркивается необходимость избавить пользователя от изучения лишнего материала, не требующегося для решения стоящих перед ним задач. Все это достигается специальной формой изложения. Большое значение придается справочному характеру документации по ПО. На протяжении всей книги отводится большое место «формату» будущих документов, их наглядности и удобству использования.

Возможно, нашему читателю покажется необходимым внести те или иные коррективы в разбираемые модели или разработать модели по другим видам документации. Надо иметь в виду, что наше отношение к обязательному составу и характеру документации по ПО поможет быть иным, чем за рубежом, так как экономические и юридические взаимодействия между авторами-разработчиками ПО и пользователями у нас иные.

Тем не менее можно надеяться, что появление перевода книги вызовет глубокий интерес к проблемам документации по ПО в широких кругах специалистов, связанных с ВТ, и будет стимулировать деятельность по стандартизации в ВТ и ее применениях.

Книга Д. Уолш не свободна от недостатков. Изложение отличается тяжелым слогом, который трудно исправить при переводе. Много повторений. По стилю изложения можно отметить отсутствие развернутых пояснений и примеров, что характерно скорее для отчета фирмы, чем книги с широким кругом читателей.

Однако большая важность и актуальность темы позволяют рассчитывать, что и в таком виде книга будет полезной, как первая среди изданий такого рода.

Перевод выполнен В. П. Варакиным, Н. В. Завгородним и М. К. Ивановым.

Н. И. Козлов

ПРЕДИСЛОВИЕ АВТОРА

Ясная, своевременная, удобная для чтения документация жизненно важна для успешного использования программного обеспечения ЭВМ. Задача подготовки такой документации для пользователей с разнообразными интересами и потребностями осложняется динамикой развития программного обеспечения и ограничениями по времени. Подход к решению этой задачи был предложен в начале 60-х годов Чарльзом П. Лехтом из корпорации АСТ. Работая с Р. Ц. Кликком из Information Systems Equipment Department at the Phoenix facility of General Electric, Лехт наметил в общих чертах подход к разработке такой документации.

Компания General Electric финансировала их исследования, в результате которых появились шаблоны документов по программному обеспечению (ПО).

Эти шаблоны были основаны на следующих простых принципах. Документация для ЭВМ разрабатывается с целью удовлетворения потребностей различных групп читателей. Системы программирования для ЭВМ неодинаковы по своей структуре и выполняют различного рода задачи по обработке информации, что также должно найти свое отражение в документации.

Наряду с этим различные категории читателей нуждаются в документации, написанной на различных уровнях (сравните, например, инструкции для оператора и инструкцию для системного программиста), в различном стиле и обладающей разной степенью детализации, что в конечном итоге определяется дееспособностью этих категорий читателей. На построение документа, его форму также влияет природа описываемой системы или ее элементов (срав-

ните, например, инструкции по различным операционным системам или инструкцию по языковому транслятору и по библиотекам).

Таким образом, для того чтобы обеспечить каждого читателя исчерпывающей информацией сообразно с его деятельностью, необходимо разработать тематический план, который отражал бы как интересы читателя, так и специфику описываемых систем программирования и их элементов.

Что представляет собой шаблон? Шаблон — это развернутый план соответствующего документа, содержащий указания по написанию каждого его пункта. Приводятся в отдельных случаях комментарии и примеры, уточняющие стиль и характер изложения пункта.

Такие шаблоны быстро доказали свое преимущество при написании удобной для чтения, хорошо организованной документации. Идея шаблона в подготовке документации по ПО оказалась плодотворной и была независимо использована рядом групп специалистов по ПО.

Фирма АСТ выразила свое отношение к шаблонам тем, что начала одной из первых их использование на практике.

Наиболее ранний набор шаблонов использовался в подготовке руководства для существовавших тогда систем программирования. Со времени разработки первых шаблонов сильно расширились аппаратные средства вычислительной техники и, соответственно, ПО. АСТ постоянно разрабатывала и развивала шаблоны для того, чтобы отразить эти изменения в оборудовании и ПО, которые настолько динамичны, что некоторая часть настоящей работы устареет, когда эта книга выйдет из печати.

Последнее не является недостатком шаблонов, так как они не жесткие формы, которых следует строго придерживаться, а, наоборот, гибкие, приспособляемые руководства. Именно по этой причине «шаблон» — это не совсем точный термин, скорее можно говорить о моделях, образцах. Поэтому в дальнейшем везде будет употребляться термин «модель», а название «шаблон» мы сохраним за самыми ранними моделями.

КАК ПОЛЬЗОВАТЬСЯ ЭТОЙ КНИГОЙ

Эта книга принесет пользу при терпеливом, систематическом использовании приводимых ниже рекомендаций. Читатель должен понимать, что модель документа по ПО — это еще не документ. Задача модели — помочь в создании исчерпывающей документации по ПО. Чтобы эта задача могла быть выполнена, читатель должен строго придерживаться следующей последовательности действий:

Прочитать «Оглавление».

Прочитать «Введение».

Перечитать «Оглавление».

Прочитать «Приложение».

Изучить модель того документа, который вы пишете.

Представить себя в роли каждого из ваших возможных читателей, чтобы подготовить самого себя к задаче написания соответствующего документа.

Скопировать часть I модели, которую вы используете.

Подготовить часть II вашего документа в соответствии с частью II модели.

Убрать заголовки, которые не применяются.

Создать новые заголовки, которые требуются.

Дать примечания в документе, чтобы можно было включить любое изменение, которое вам может потребоваться.

Примечание. В «Приложении» содержатся подробные систематические указания по созданию документа с помощью модели.

ВВЕДЕНИЕ

Назначение моделей документации по программному обеспечению. Модели документации для подготовки инструкций по программному обеспечению ЭВМ разрабатываются для того, чтобы:

- * облегчить задачу организации материала в каждом типе документа;
- * описать уровень и стиль изложения соответствующего документа, а также предполагаемое его использование.

Модели документов в значительной степени облегчают труд по созданию инструкций для ПО и, что остро чувствует пользователь, оказывают положительное влияние на качество документов.

Что дает использование моделей читателю. Если модели используются правильно, то документация, составленная с помощью моделей, отличается:

- * исчерпывающим содержанием;
- * исчерпывающим «Оглавлением»;
- * логической завершенностью изложения, вытекающей из самого порядка разработки документа;
- * способностью оказать действенную помощь читателю в его работе.

Что дает использование моделей автору. Если модели используются правильно, то автор документа уверен:

- * в том, что оглавление, приведенное в модели, поможет ему избежать потери важной информации;
- * в руководстве по организации и стилю изложения материала;

- * в том, что модель позволяет удовлетворить широкому спектру требований к документам, описывающим различного рода системы ПО.

Ответственность автора. Наряду с облегчением конторского и организационного бремени человека, который пишет руководство, использование моделей увеличивает его ответственность за создаваемую на их основе документацию.

- * Он освобождается от необходимости организовывать структуру материала. Следовательно, он должен более полно посвятить себя деятельности по подготовке удобной для чтения, полезной документации.
- * Автор должен войти в контакт с каждым из своих возможных читателей. Он должен, следовательно, говорить с каждым читателем на его языке и давать нужную ему информацию.
- * При подготовке инструкций по программному обеспечению существенным является то, что читатели данного документа могут значительно различаться в знаниях, в интересах. Модель помогает автору выделять группы читателей, но ответственность за правильное и понятное изложение материала для каждой группы читателей должна лежать на создателе документа.
- * Иногда нужная форма подачи материала поддается структуре, рекомендованной для данного пункта документации по программному обеспечению.

Разработка моделей. Разработчики документов по программному обеспечению ЭВМ сталкиваются с проблемами, связанными:

- * с полнотой содержания;
- * с логическим построением и использованием документа;
- * с формой представления, стилем и уровнем изложения и т. д.

Эти проблемы присущи не только области обработки информации на ЭВМ. Если, например, в качестве читателя рассматривать специалиста-производственника, а вместо программного обеспечения

ЭВМ — промышленные агрегаты, с которыми этот специалист связан по роду своей производственной деятельности, то приведенные выше рассуждения относительно разработки документации в той или иной мере будут справедливы и в этом случае.

Проблема создания документации существует во многих отраслях, однако своеобразные условия обработки информации на ЭВМ делают наличие качественной документации по программному обеспечению особенно важным.

Задача создателя документов усложнена постоянно меняющимся составом аппаратуры и средств ПО, потребностью абсолютно точного, подробного в нужной степени изложения, необходимостью внесения в документы по возможности минимальных изменений, которые обусловлены развитием средств аппаратуры и ПО, и, наконец, тем, что читатели разных категорий и разных уровней используют одни и те же типы документов (например, инструкции по операционной системе, которые могут быть предназначены как оператору, так и пользователю-прикладнику).

Модели призваны облегчить эту задачу и дают ответы на некоторые из вопросов, возникающих при разработке документации по ПО.

Первоначально разработка такой документации поручалась авторам-разработчикам ПО. Этим достигалась полнота охвата и логическая завершенность изложения материала, так как разработчик лучше, чем кто-либо, знаком с продукцией своей деятельности и ее характеристиками. Ориентация на потребности читателя позволяла совершенствовать созданную авторами документацию по ПО. Однако вскоре стало ясно, что такой путь создания документации не свободен от серьезных недостатков. Полное решение проблемы создания документации дают модели.

Шаблоны — самые ранние модели. Модели были первоначально созданы именно как шаблоны. Они составляли часть рабочих средств программиста наряду с шаблонами для изображения блок-схем. Поэтому первые модели были названы шаблонами.

Первые модели были разработаны, построены, написаны и проверены сотрудниками АСТ. После

первых успехов в АСТ применение этих моделей охватило все типы деятельности по программированию в нескольких странах.

Программисты-системщики, создающие программное обеспечение для фирм, изготавливающих ЭВМ, использовали модели, чтобы документировать это программное обеспечение. Программисты, подготавливающие прикладные программы-системы для решения соответствующих прикладных задач, также документировали свои системы в соответствии с моделями.

Сами модели были переведены на французский язык; документы, полученные на их основе, были переведены с английского и французского на итальянский и немецкий языки.

Развитие форм моделей. Развивалась вычислительная техника, развивались и модели. Но это развитие происходило не обособленно. Развивающаяся техника стимулировала разработку моделей, а модели влияли на развитие техники.

Развитие техники привело к изменению содержания документов, а возросшая сложность и возможности оборудования привели к более разностороннему и сложному программному обеспечению. Рост в этой области вызвал сдвиг в оценке роли документации по ПО и заставил подойти к задаче создания документов с большей ответственностью. Широкое использование моделей способствовало их совершенствованию. Все нововведения в области разработки и использования ЭВМ находили свое отражение в моделях.

Современная форма моделей. На данном этапе развития представленные здесь модели документации дают рекомендации для создания документации по ПО.

Разумеется, набор моделей, включенных в эту книгу, не описывает всего многообразия документов по ПО. В книгу включены лишь наиболее отработанные модели документов.

Каждая из приведенных моделей достигла своей современной формы после многократного применения на практике, усовершенствования, поэтому они

представляют собой проверенные руководства по написанию соответствующих документов.

Важной особенностью моделей является их гибкость. Усовершенствования прямо обусловлены трудностями применения модели к нововведениям в ПО, трудностями включения новой информации в документ.

Модели, следовательно, представлены как фундаментальные руководства, как исходные пункты для подготовки документов по программному обеспечению.

Будущее моделей. Техника счета на ЭВМ не прекратила развиваться. Нельзя также считать, что процесс создания документации в рамках существующих моделей отработан до совершенства и что данный набор моделей охватывает все возможные типы документов. Следовательно, модели нужно совершенствовать. Предполагается, что в их развитие и становление внесет свой посильный вклад сам пользователь, который будет дополнять и уточнять их в процессе практического применения и сообщать о всех вносимых им принципиальных изменениях.

Введение усовершенствований в модели облегчается формой их представления.

Используемая форма представления. Для представления моделей были выработаны определенные правила. Они призваны облегчить использование моделей и дать средства, позволяющие легко разобратся в различных видах информации, содержащейся в документах.

Каждая модель представляется в виде двух основных частей:

- * Часть I — преимущественно оглавление документа, который будет написан по модели.
- * Часть II — подробное систематическое описание подготавливаемого документа в соответствии с оглавлением и структурой информации.

Для того чтобы облегчить работу с моделью, ее составители придерживались следующих правил:

- * Основные заголовки и подзаголовки как в части I, так и в части II выделяются
- * Образцы предложений, которые используются для иллюстрации описания документа и которые

могли бы составлять часть самого документа, также выделяются.

Когда описание документа (часть II модели) подготовлено в виде рукописи:

- * выделяются разделы, которые используются в виде смысловых разделителей;
- * делаются ударения на особо важные моменты в каждой отдельной теме;
- * разделы и примеры нумеруются в каждой теме.

Как уже говорилось, модели постоянно развивают и приспособливают к условиям, в которых применяются документы. Примером такой приспособляемости может служить то, что модели, приведенные в этой книге, по своей структуре несколько отличаются друг от друга. В одних, например, нет тех рекомендаций, которые есть в других. Это же самое относится и к некоторым заголовкам.

Использование моделей. Предполагается, что создатель документов будет пользоваться моделями, учитывая рекомендации по их эффективному использованию, изменяя их так, чтобы они соответствовали поставленной перед ним задаче. Во всяком случае предполагается, что он:

- * включит в документ все необходимые заголовки, показанные в части I;
- * вычеркнет заголовки, которые не применяет (сохраняя структуру информации);
- * добавит новые заголовки, если это необходимо;
- * подготовит документ в соответствии с руководством (описанием документа), которое дано в части II.

Примечание. Модели представляют собой руководства. Они несколько уменьшают ответственность пользователя за подготовку оглавления и плана, а также дают некоторую ориентацию для стиля. Однако при этом увеличивается ответственность за качество создаваемой документации.

Использование моделей должно благоприятствовать появлению более содержательных документов, написанных в удобном для чтения стиле и построенных так, чтобы их легко было использовать,

ОБЩИЕ РАЗДЕЛЫ

Каждая из моделей содержит четыре стандартных раздела, которые описаны здесь в общих словах. Этими разделами являются:

Предисловие.

Оглавление.

Рекомендации по эффективному использованию документа.

Словарь терминов.

То, что говорится ниже, применимо к этим разделам для любого документа и поэтому поясняется один раз.

Предисловие. По определению «Предисловие» является введением в работу. Его цель заключается в том, чтобы сказать, для кого предназначена книга, какую информацию она дает и как она должна быть использована.

Документация по программному обеспечению ЭВМ обычно представляет собой информацию, разбитую в соответствии с группами читателей, чтобы удовлетворить потребностям возможно более широкого круга.

Каждая отдельная группа читателей должна получить документ на языке, отражающем ее собственные потребности, и должна использовать документ так, чтобы он послужил ей наилучшим образом.

По этим соображениям «Рекомендации по эффективному использованию» заменяют «Предисловие».

«Предисловие» может, следовательно, быть опущено по усмотрению автора.

Однако, наряду с «Рекомендациями по эффективному использованию», автор может подготовить и «Предисловие», чтобы дать в нем некоторую особо важную информацию о документе.

Необходимо следить за тем, чтобы вводные замечания ограничивались общей характеристикой документа, а не излагали суть дела. Полная информация об объекте рассмотрения должна быть включена в соответствующий раздел подготавливаемого документа.

Оглавление. Оглавление для каждого документа дается в части I модели этого документа.

Оглавление предназначено для того, чтобы:

- * дать читателю картину организации документа,
- * обеспечить читателя действенным средством для поиска нужной ему информации в документе.

Об этих целях следует помнить при подготовке «Оглавления» для каждого документа.

Начинайте с перечисления главных подзаголовков, приведенных в части I модели документа, которую вы используете. Затем добавьте главные заголовки и необходимые подзаголовки, чтобы удовлетворить требования, предъявляемые к данному документу. Нумеруйте все рисунки и таблицы и включайте их в «Оглавление» соответствующего раздела. Добавьте дополнительный список таблиц и рисунков.

В «Оглавлении» в скобках указывается номер страницы составляемого документа.

Рекомендации по эффективному использованию документа. Как отмечалось выше, этот раздел документа должен давать представление об организации документа и облегчать процесс его чтения благодаря ориентации на ту или иную группу читателей.

Можно указать определенные общие принципы подготовки «Рекомендаций по эффективному использованию документа». Последовательность действий по их подготовке рассматривается ниже.

Если документ представляет интерес для различных групп читателей, укажите наиболее важные разделы для каждой из таких групп.

Очень важно, чтобы документ, написанный для определенного круга читателей, использовался ими в соответствии с замыслом автора. Если этого не добиться, то даже хорошо написанный и содержательный документ может оказаться бесполезным.

«Рекомендации по эффективному использованию документа» должны, с одной стороны, помочь читателю выделить необходимый ему материал, избавляя его от изучения лишнего, а с другой стороны, помочь максимально использовать весь имеющийся в документе материал.

«Рекомендации» должны отвечать на следующие вопросы о документе:

- * кто является предполагаемым читателем, его образование и нужды;
- * что содержит документ и какая информация может быть получена из него;
- * в каких случаях его лучше всего использовать;
- * зачем документ был написан, каковы цели, которым он служит;
- * как найти различные элементы информации, содержащейся в нем;
- * как наиболее эффективно пользоваться документом в различных случаях его использования.

Ответы на эти вопросы могут содержаться в кратких пояснениях. В отдельных случаях удобно использовать табличную форму. Ответы на вопросы «Как использовать документ?» и «Где найти специфические элементы информации?» лучше всего представляются в виде последовательности шагов.

- * Например, можно использовать блок-схему, подобную блок-схеме программы.

Особо подчеркните важность «Оглавления». Ссылайтесь на имеющиеся «Приложения». Укажите на возможность использования «Словаря терминов».

Словарь терминов. Модели, представленные здесь, предназначены для использования при подготовке ряда документов, описывающих обширную систему ПО или часть некоторой совокупности материалов по ПО. Поэтому целесообразно снабдить документ словарем терминов, стандартным для упомянутой системы или совокупности. В каждом документе в такой словарь включаются термины, используемые в этом документе. Включенные в словарь термины должны обязательно определяться.

При отсутствии общепринятого словаря подготовьте ваш собственный словарь.

МОДЕЛИ ДОКУМЕНТАЦИИ

Модель 1

ПОДГОТОВКА ОПИСАНИЯ ТЕХНИЧЕСКОГО ЗАДАНИЯ НА ПРОЕКТ СИСТЕМЫ/ПРОГРАММЫ

Часть I

Предисловие (общий раздел) ().

Рекомендации по эффективному использованию описания ().

Оглавление (общий раздел) ().

Раздел А: Постановка задачи ().

Раздел В: Основополагающая информация ().

Раздел С: Входные данные

I. Запись (тип) ().

a. Использование.

b. Формат.

c. Тип.

d. Размер.

} Повторите для каждого типа записи. Следующий тип нумеруется римской цифрой II.

Раздел D: Выходные данные ().

I. Запись (тип) ().

a. Использование.

b. Формат.

c. Тип.

d. Размер.

e. Носитель.

f. Хранение.

g. Предписание.

} Повторите для каждого типа записи. Следующий тип нумеруйте римской цифрой II.

Раздел E: Автоматизированная обработка ().

I. Описание ().

II. Блок-схема ().

III. Вспомогательная информация ().

IV. Ограничения ().

V. Требуемая точность ().

VI. Необходимое оборудование ().

- VII. Исправление ошибок ().
- VIII. Специальные соображения ().
- Раздел F: Обработка на электромеханическом устройстве и ручная обработка ().
- I. Описание ().
- II. Блок-схема ().
- III. Вспомогательная информация ().
- IV. Ограничения ().
- V. Требуемая точность ().
- VI. Необходимое оборудование ().
- VII. Исправление ошибок ().
- VIII. Специальные соображения ().
- Раздел G: Критерий полноты ().
- Раздел H: Оценка времени ().
- Раздел I: Оценка стоимости ().
- Раздел J: Оправданность ().
- Раздел K: Отчеты о проделанной работе ().
- Библиография ().
- Состав персонала ().
- Словарь терминов (общий раздел) ().

Часть II

Рекомендации по эффективному использованию описания

При объяснении цели и использования «Описания технического задания на проект системы/программы» следует иметь в виду следующее:

- * Эта модель предназначена для использования персоналом, занятым инженерными или коммерческими расчетами, и персоналом по обработке данных. Ею следует руководствоваться при написании технического задания на проект системы/программы, разрабатываемой и внедряемой персоналом по обработке данных.
- * Степень детализации, устанавливаемая в модели, должна быть выдержана в документе так, чтобы ресурсы (люди, машины и время) могли быть точно оценены в проекте.
- * Описание должно давать достаточную детализацию для проведения планирования.
- * Уделяйте особое внимание точности и полноте описаний.

- * Обращайтесь к организации спецификаций, предлагаемой в модели, и придавайте разделам документа законченный вид, снабжая их различного рода примерами, уточнениями, которые позволят читателю лучше понять и уяснить поставленную перед ним задачу.
- * В частности, дайте естественное сравнение ввода и вывода, сопоставьте исходную информацию с предполагаемыми результатами.
- * Поощряйте читателя в разработке и развитии схем.

Раздел А: Постановка задачи

В этом разделе дайте общую постановку задачи, объяснения, какие вычисления, функции и/или процедуры должны выполняться разрабатываемой программой/системой на ЭВМ для того, чтобы она могла реализовать соответствующее ей коммерческое, инженерное применение *).

- * Этот раздел следует писать так, чтобы его содержание было понятно даже тем, кто не знаком с функциями и их реализацией в такой системе/программе.

Раздел В: Основополагающая информация

Объясните цель применения, для автоматизации которого создается данная система/программа. Проследите его историческое развитие.

Если существуют применения, подобные рассматриваемому, и они уже автоматизированы, расскажите о них.

Опишите, в общих словах, методы, используемые при автоматизации рассматриваемого применения.

Отсылайте читателя к «Библиографии» по данному вопросу, которая более подробно освещает вопросы, связанные с данным применением.

- * Нарисуйте общие диаграммы или, если необходимо, набор диаграмм, чтобы проиллюстриро-

*) Здесь «применение» — область приложения программы/системы.

вать функции и/или процедуры, их взаимосвязи, последовательность, в которой они появляются, и, если это возможно, последовательность «бумажной» работы и/или работы с данными от начала до конца применения.

- * Рисуйте диаграммы для каждого отдельного случая, если единая иллюстрация невозможна.
- * Установите, будет ли разрабатываемая программа/система конечной или промежуточной версией, и опишите границы этой программы/системы.
- * Если это уместно, опишите указанные вопросы отдельно.

Примечание. Если вы описываете ряд программ, повторите разделы от С до F для каждой программы.

Раздел С: Входные данные

1. Запись (тип). Этот раздел должен быть повторен для каждого типа записи.

Примечание. Большая часть информации, предусмотренной в этом разделе, связана с формой записи для коммерческих применений. Когда эта форма уже используется на практике и лишь воспроизводится в этом разделе, необходимо дать дополнительную информацию лишь по тем вопросам, которые не охватываются этой формой.

а. *Использование*. Объясните, какой цели служит запись, т. е. для чего она используется.

б. *Формат*. Проиллюстрируйте формат входной записи, а также формат всех блоков или файлов, по которому могут быть организованы группы записи.

Будьте точны при указании всех значащих областей (полей) и подобластей (подполей), которые содержатся в записи.

Если формат данных неодинаков для коммерческих и инженерных расчетов, то его следует описывать под разными заголовками.

Если входные данные не организованы по специальному формату, формат должен быть разработан и установлен персоналом по обработке данных.

с. Тип. Укажите, включают ли данные только буквенные символы, только числовые символы или комбинации и тех и других.

* Если числа включаются, укажите, являются ли они десятичными, двоичными, восьмеричными, шестнадцатеричными или числами другого вида.

* Специальные символы или символы, которые будут появляться, также должны быть указаны.

d. Размер. Выражайте в количестве символов или в других подходящих единицах длину входной записи данных.

Раздел D: Выходные данные

1. *Запись* (тип). Этот раздел должен быть повторен для каждого типа записи, которая будет вырабатываться программой.

Примечание. Большая часть информации, предусмотренной в этом разделе, появляется в форме схемы записи коммерческих применений.

a. Использование. Установите, какой цели служит запись, т. е. для чего она используется.

Если часть выходной записи должна быть скомпонована и напечатана в форме отчета, объясните, для чего используется отчет, и кто будет соответствующим пользователем.

b. Формат. Нарисуйте иллюстрацию схемы или формата любой выходной записи или отчета, которые должны быть получены.

Если требуется более чем один формат, давайте иллюстрацию каждого.

Если напечатанное сообщение (отчет) включается в документ, будьте точны при указании того, какие строки заголовков, общие строки, строки надписей и завершающие строки должны быть в него включены.

Отмечайте, должно ли выполняться какое-либо редактирование или модификации выходных данных, и, если это так, укажите степень и тип редактирования и модификации.

с. Тип. Указывайте, должны ли выходные данные состоять только из буквенных символов, только цифр или комбинации и того и другого.

- * Если числа включаются в выходные данные, отмечайте, будут ли они в десятичном или другом представлении.
- * Укажите все специальные знаки или символы, которые используются в выходных данных.
- d. **Размер.** Выражайте числом символов или в других подходящих единицах длину записей, которые составляют выходные данные.
 - * Если выходные записи должны быть сгруппированы вместе в блоки и/или файлы, указывайте число записей в каждом блоке и/или файле.
 - * Также определите, постоянен ли размер выходной записи блока или файла. Если он переменный, то укажите возможный диапазон его изменения.
- e. **Носитель.** Укажите, какой носитель должен быть использован для хранения выходных данных.
 - * Например, должны ли они храниться на перфокартах, на перфоленте, магнитной ленте или в печатном виде.
- f. **Хранение.** Укажите, как долго выходные данные, которые хранятся на перфокартах, магнитной ленте и т. д., должны сохраняться, и какие меры должны быть приняты для того, чтобы иметь резервные копии данных в случае, если копия при использовании портится.
- g. **Предписание.** Укажите, какое предписание должно быть сделано относительно выходных данных.
 - * Например, если печатное сообщение (отчет), выданное программой данного применения, должно быть направлено в отдел платежных ведомостей, то читатель должен быть информирован об этом.

Раздел Е: Автоматизированная обработка

Дайте в этом разделе подробное пошаговое описание каждого вычисления, функции и/или процедуры, которая должна быть выполнена проектируемой программой для ЭВМ.

Представьте эту информацию в терминах, понятных проектировщику, который способен схватить существо применения, но не знаком детально с соответствующей областью.

Если вы представите эту информацию ясно и подробно во всех отношениях, профессиональный прикладной программист сможет составить программу непосредственно с помощью этого раздела.

I. Описание. Опишите подробно каждое вычисление, функцию и/или процедуру, которые должны быть автоматизированы с помощью ЭВМ.

В процессе этого описания будьте внимательны при связывании каждой ступени обработки с материалом, представленным в разделе С «Входные данные» и разделе D «Выходные данные», т. е. указывайте, какие входные данные обрабатываются на каждой ступени и какие выходные данные получаются в результате этой обработки.

II. Блок-схема. Рисуйте блок-схему, показывающую основные ступени обработки, включаемые в применение, и последовательность, в которой эти ступени выполняются.

III. Вспомогательная информация. Кратко расскажите о тех теоретических положениях, методах и предположениях, которые используются при автоматизации данного применения.

IV. Ограничения. Определите все ограничения, налагаемые характером или использованием вычислений, функций и/или процедур, которые должны быть автоматизированы с помощью ЭВМ.

V. Требуемая точность. Указывайте требуемую степень точности любого математического или статистического вычисления, которое должно быть автоматизировано с помощью ЭВМ.

VI. Необходимое оборудование. Перечислите оборудование, необходимое, чтобы эффективно реализовать автоматизируемое применение.

VII. Исправление ошибок. Опишите основной метод или технические приемы, которые будут использоваться в проектируемой программе для исправления любых ошибок, которые могут появиться.

VIII. Специальные соображения. Ознакомьте читателя с теми соображениями, которых не касались нигде в вашем описании и которые программисту следует осознать перед тем, как он фактически начнет писать программу.

Раздел F: Обработка на электромеханическом устройстве и ручная обработка

Расскажите о тех вычислениях, функциях и/или процедурах, которые должны быть выполнены без использования ЭВМ (например, перфорация, накопление данных и т. д.). Чтобы не повторяться, ссылайтесь на заголовки и объяснительный текст раздела E «Автоматизированная обработка», чтобы не дублировать информацию, относящуюся к другим разделам.

Помните, что ссылки на программы и ЭВМ в разделе E следует заменить ссылками на ручные процедуры и оборудование в описании настоящего раздела.

Раздел G: Критерий полноты

Определите, что будет пониматься под полнотой проекта. Дайте методы проверки и другие подобные критерии.

Раздел H: Оценка времени

Оцените количество времени, которое займет осуществление проекта. Будьте внимательны при включении количества времени, требуемого для разработки, программирования и проверки проектируемой программы/системы.

Раздел I: Оценка стоимости

Оцените стоимость проекта. Различайте затраты на персонал и затраты на оборудование, стоимости разработки и эксплуатации.

Раздел J: Оправданность

Укажите здесь достоинства проекта.

Раздел K: Отчеты о проделанной работе

Установите частоту и формы отчетов о проделанной работе. Если нужно, включите в документ список пунктов, куда эти отчеты должны направляться.

Библиография

Приведите авторов и названия книг, статей, сообщений и т. д., которые содержат информацию, дополняющую вычисления, функции и/или процедуры, включенные в применение, которое должно быть автоматизировано.

Состав персонала

Перечислите весь дополнительный персонал, который может потребоваться персоналу по обработке данных, или тот персонал, который будет внедрять проект применения.

Включите звание или должность каждого сотрудника и дайте краткое перечисление его обязанностей, функций и поля деятельности, связанных с проектом.

Укажите тех, от кого может быть получена как запланированная, так и незапланированная помощь.

Модель 2

ПОДГОТОВКА РУКОВОДСТВА ПОЛЬЗОВАТЕЛЮ ПО РАБОТЕ С НЕДИАЛОГОВОЙ СИСТЕМОЙ/ПРОГРАММОЙ

Часть I

Предисловие (общий раздел) ().

Рекомендации по эффективному использованию этого руководства ().

Оглавление (общий раздел) ().

Раздел А: Введение ().

I. Назначение ().

II. Выполняемая работа ().

III. Ограничения ().

Раздел В: Как использовать (наименование системы/программы)

I. Описание входных данных ().

II. Описание работы ().

III. Подготовка запросов на обработку ().

IV. Интерпретация получаемых выходных данных ().

V. Исправление ошибок и повторная обработка ().

Словарь терминов (общий раздел) ().

Часть II

Рекомендации по эффективному использованию этого руководства

При описании назначения и использования «Руководства пользователю по работе с недиалоговой прикладной системой/программой» должна быть дана следующая информация:

- * Недиалоговая прикладная система/программа является системой ПО для ЭВМ или программой, которая выполняется без непосредственного вмешательства как пользователя, так и операционной системы, которой она обрабатывается.
- * Читатель этого документа является представителем персонала, который будет использовать прикладную систему/программу.
- * Содержание этого «Руководства» включает в себя ответы на вопросы:
 1. Какие входные данные должны быть введены в систему/программу.
 2. Как входная информация вводится.
 3. Каковы возможности системы/программы.
 4. Какая выходная информация получается в результате работы системы/программы.
 5. Как выходная информация должна восприниматься и использоваться.

Снабдите ваше «Руководство» точными инструкциями для нахождения такой информации.

Раздел А: Введение

Этот раздел «Руководства» должен давать пользователю всю основополагающую информацию, которая ему требуется, чтобы понять, что представляет собой прикладная система/программа и что она способна делать.

- * Пишите этот раздел на языке, понятном пользователям.
- * Помните, что большинство пользователей не будет хорошо знакомо с данной прикладной системой/программой или вообще ни с одной похожей на нее системой/программой.

- * Используйте техническое задание в качестве исходного материала.

Короче говоря, не упускайте ничего. Начинайте с того, что, возможно, покажется наиболее элементарным, и ведите читателя к наиболее сложной информации, которую он должен понять.

- * Сообщите здесь читателю, что он может пропустить этот раздел, если он хорошо знаком с информацией, представленной здесь.

I. Назначение. Объясните четко и кратко назначение прикладной системы/программы с точки зрения поставленной задачи.

II. Выполняемая работа. Перечислите и кратко объясните все функции, процедуры или обработку, которые прикладная система/программа выполняет в установленном порядке или по выбору пользователя.

Будьте внимательны, когда указываете, что выполняется в установленном порядке, а что выполняется по выбору.

Не вдавайтесь в подробности в этом пункте. Эта информация обсуждается более подробно в разделе В под заголовком «Описание работы».

III. Ограничения. Установите, что ваша прикладная система/программа не может делать.

- * Например, если это программа, которая может обрабатывать только внутренние арендные договоры, подчеркните, что она не может обрабатывать иностранные арендные договоры.

Раздел В: Как использовать (наименование системы/программы)

I. Описание входных данных. Опишите каждый тип входной записи данных, которые пользователь должен предоставить вашей прикладной системе/программе.

- * Установите назначение каждого типа или категории.
- * Нарисуйте картину, изображающую каждый тип. Для коммерческой системы/программы просто приведите пример формы полной записи. Приведите реальные примеры записей каждого типа.
- * Приведите пример записей каждого типа или категории.

- * Для каждого типа опишите назначение всех полей или подполей, составляющих записи.
- * Укажите длину записей (в символах или в других подходящих единицах), относящихся к каждому типу. Если она переменная, укажите диапазон изменения.
- * Укажите, может ли запись, относящаяся к каждому типу, содержать двоично-десятичные, двоичные, восьмеричные, шестнадцатеричные и другие данные.

Если записи содержат двоично-десятичные данные, то укажите, включают ли они в себя только буквенные символы, цифры или комбинацию и тех и других.

Будьте внимательны при указании того, какие специальные знаки и/или символы могут появляться во входной записи. Чтобы описать входные записи данных, создайте подраздел о самих формах (бланках).

- * Установите назначение каждой формы.
- * Приведите рисунок каждой формы.
- * Укажите, пишется ли информация в одну строку, две строки и т. д. каждой формы, составляющей входные данные.
- * Разбейте данные, которые пользователь помещает в строке каждой формы, на категории и рассмотрите каждую категорию с точки зрения того, в каких позициях строки помещаются данные и каково назначение этой категории данных, т. е. после установления категории данных объясните каждую категорию так же, как если бы она была операндом или параметром оператора языка.

Если пользователь может включать управляющие записи в поток входных данных, то опишите их в этом подразделе.

II. Описание работы. Опишите шаги работы прикладной системы/программы в заранее определенном порядке, а также любые варианты обработки различных типов входных записей, которые может представлять пользователь.

- * Подготовьте информацию, представленную здесь, для описания входных записей данных под заголовком «Описание входных данных». Укажите, как каждый шаг обработки изменяет данные.

- * Если работа программы меняется в соответствии с каждым типом входных записей, отметьте это и объясните, почему и как она меняется.
- * Если последовательность, в которой выполняются шаги работы, является определяющей, дайте блок-схему, которая показывает последовательность, в которой выполняется каждый шаг обработки.
- * Не приводите здесь подробности обработки или фактическую внутреннюю работу системы/программы. Этот тип информации относится к «Инструкции по внутренней логике».
- * Покажите, как пользователь должен указывать варианты обработки, если таковые имеются.

Если выбор обработки делается с помощью операторов, управляющих программой:

опишите каждый оператор;
нарисуйте картинку формата каждого оператора и объясните любой параметр, который может быть включен в каждый оператор.

Если последовательность обработки в пределах программы меняется в соответствии с выбором пользователя, то:

установите, что это имеет место;
укажите, как обработка меняется при выборе каждой возможности; нарисуйте, если это необходимо, общую блок-схему, иллюстрирующую каждое возможное изменение.

III. Подготовка запросов на обработку. Опишите, как пользователь подготавливает запрос на обработку.

- * Если он должен заполнить соответствующую форму (бланк), проиллюстрируйте эту форму и покажите, как она должна заполняться.
- * Укажите, куда должен направляться этот запрос.
- * Сообщите, какие записи входных данных и какие заказы на печать отчетов и/или специальные управляющие карты для оборудования должны прилагаться к запросу.
- * Если существует фактор времени, укажите дату или время, когда эта информация должна направляться в отдел или службы, которые контролируют выполнение запросов.

- * Включите в документ подробную памятку, позволяющую пользователю проконтролировать самого себя при составлении запроса.

IV. Интерпретация получаемых выходных данных.

Наиболее общий тип выходных данных, которые пользователь получит после того, как работа выполнена, является сообщением (отчетом) или последовательностью сообщений.

Объясните подробно, как пользователь должен интерпретировать получаемые сообщения.

- * Установите назначение каждого сообщения.
- * Проиллюстрируйте формат каждого сообщения.
- * Приведите пример каждого сообщения.
- * Разделите по категориям информацию, которая содержится в каждом сообщении (например, согласно с данными, которые появляются под каждым заголовком, названием, в столбце или группе столбцов). Затем объясните значение каждой категории информации.
- * Предлагайте возможные пути использования информации, содержащейся в каждом сообщении.

Если выходная информация, которую получает пользователь, состоит преимущественно из выходных записей данных, опишите каждый тип выходной записи данных, охватывая такие же пункты, которые вы обычно используете при описании входных записей данных.

- * Будьте внимательны при указании того, как организуются записи в блоки и/или файлы или наборы данных.
- * Если эти записи объединяются в более крупные логические группы, такие, как блоки или файлы, опишите каждую группу с точки зрения ее назначения и формата. Дайте признаки начала и конца группы.

V. Исправление ошибок и повторная обработка.

Определите все процедуры, которым должен следовать пользователь для определения точности или обоснованности получаемых выходных данных.

Если пользователь получил неточные или ошибочные выходные данные, объясните, как он должен сообщить об этом персоналу по соответствующим работам и как перейти к повторной обработке.

Если существуют способы проверки входных данных, объясняющие ошибки в выходных данных, то опишите здесь эти способы.

Модель 3

ПОДГОТОВКА РУКОВОДСТВА ПО РАБОТЕ С НЕДИАЛОГОВОЙ ПРИКЛАДНОЙ СИСТЕМОЙ/ПРОГРАММОЙ

Часть I

- Предисловие (общий раздел) ().
- Рекомендации по эффективному использованию этого руководства ().
- Оглавление (общий раздел) ().
- Раздел А: Общая информация ().
 - I. Введение ().
 - II. Состав программного обеспечения ().
 - a. Операционная система ().
 - b. Прикладная система/программа ().
 - III. Требования к оборудованию ().
 - a. Операционная система ().
 - b. Прикладная система/программа ().
- Раздел В: Подготовка задачи ().
 - I. Управляющие директивы для операционной системы ().
 - II. Входные данные пользователя ().
 - III. Управляющие директивы для оборудования и прикладной системы/программы ().
 - IV. Организация входного потока ().
 - V. Требования к планированию ().
- Раздел С: Выполнение задачи ().
 - I. Начало ().
 - II. Выполнение ().
 - III. Прерывание ().
 - IV. Восстановление ().
 - V. Завершение ().
 - a. Нормальное ().
 - b. Ненормальное ().

Раздел D: Выходные данные ().
I. Данные пользователя ().
II. Организация данных ().
Приложение: Сообщения оператору ().
Словарь терминов (общий раздел) ().

Часть II

Рекомендации по эффективному использованию этого руководства

При описании цели и использования этого «Руководства» должна быть дана следующая информация:

- * рабочий персонал включает в себя: персонал по внешним устройствам, диспетчерский персонал, персонал по управлению данными и операторов ЭВМ;
- * деятельность по обработке информации такого персонала включает в себя все шаги и действия, которые должны быть выполнены с момента получения данных от пользователя и до момента, когда пользователь получает свои выходные данные.

Удостоверьтесь, что «Рекомендации по эффективному использованию» отражают потребности всех перечисленных групп рабочего персонала и могут послужить им в качестве путеводителя по документу.

Раздел A: Общая информация

I. Введение. Опишите, в общих чертах, назначение, возможности и предполагаемое использование прикладной системы/программы.

Это описание может быть просто сокращенным изложением информации, которая находится в разделе I «Введение».

Тех, кто интересуется полным подробным описанием, отсылайте к «Руководству пользователю по недиалоговой системе/программе».

Подкрепите это описание диаграммой, которая показывает различные виды ввода и вывода, осуществляемые прикладной системой/программой.

II. Состав программного обеспечения.

а. Операционная система. Укажите операционную систему, в которой будет работать прикладная система/программа.

Перечислите компоненты этой операционной системы, которые требуются, чтобы выполнить задачи, связанные с данной системой/программой.

* Если все прикладные системы/программы используют одинаковые компоненты операционной системы, укажите инструкцию, которая должна использоваться в случае, если при выполнении задач возникнут вопросы.

* Если эта конфигурация нестандартна, указывайте, где и как нужно обращаться к каждой компоненте, каковы используемые накопители.

б. Прикладная система/программа. Опишите прикладную программу или каждую программу в этой прикладной системе. Укажите, где и как нужно обращаться к ним.

Если для различных вариантов обработки требуются различные программы или конфигурации программ, укажите, что это имеет место, и дайте таблицу, которая показывает, какое программное обеспечение требуется для каждого варианта.

III. Требования к оборудованию.

а. Операционная система. Определите конфигурацию оборудования ЭВМ и дополнительного оборудования, требуемого операционной системе, в которой прикладная система/программа должна будет работать.

* Если конфигурация стандартная, просто укажите инструкцию или стандарт, в которых может быть найдено описание этой конфигурации.

б. Прикладная система/программа. Определите конфигурацию оборудования и вспомогательных устройств, требуемых для выполнения задачи, связанной с прикладной системой/программой.

* Если вы работаете с операционной системой, укажите только оборудование, требуемое в дополнение к тому, которое требуется самой операционной системе.

- * Если вы не работаете с операционной системой, перечислите все оборудование, которое требуется, чтобы выполнить задачу.
- * Если эти требования меняются в соответствии с используемыми вариантами обработки, скажите об этом и дайте таблицу, которая показывает эти варианты и рядом с ними требования к оборудованию.

Раздел В: Подготовка задачи

Объясните подробно, что должно быть сделано, чтобы подготовить к выполнению задачу, связанную с прикладной системой/программой, с момента получения задачи от пользователя и до момента получения ее оператором ЭВМ.

I. Управляющие директивы для операционной системы. Опишите управляющие директивы для операционной системы, которые требуются для того, чтобы выполнить задачу, связанную с прикладной системой/программой. Перечислите эти директивы в том порядке, в котором им следует появляться при вводе задачи.

Установите специфическое назначение каждой.

Включите в документ иллюстрацию того, как выглядит каждая директива, когда она подготовлена.

Объясните причину включения каждого значащего параметра или операнда в каждую директиву.

Если директивы входят в стандартный набор директив, дайте ссылку на документ, где можно найти описание этих директив.

II. Входные данные пользователя. Перечислите каждый тип или категорию входных данных пользователя.

Опишите общее назначение каждого типа.

Нарисуйте схему, которая иллюстрирует формат каждого типа данных.

Дайте пояснения к формам (бланкам), если таковые имеются, на которых пользователь должен описывать входные данные.

Дайте пример такой формы.

Включите, если это необходимо, все специальные инструкции, которые перфораторщики должны

знать при перенесении входных данных пользователя на перфокарты.

Опишите автономную обработку, которая должна быть выполнена над входными данными пользователя.

Установите, когда, как и куда входные данные пользователя должны быть переданы.

III. Управляющие директивы для оборудования и прикладной системы/программы. Опишите все-управляющие директивы, требующиеся:

1) оборудованию, на котором задача должна быть выполнена, или

2) прикладной системе/программе.

* Например, карта DATE (ДАТА) могла бы потребоваться в предыдущем пункте 1), в то время как для пункта 2) могли бы описываться варианты обработки.

Ссылайтесь на раздел I «Управляющие директивы для операционной системы» для описания информации, которая должна появляться в каждой такой директиве.

IV. Организация входного потока. Опишите требуемый состав входного потока задачи, связанной с прикладной системой/программой.

* Укажите, какие управляющие директивы, данные пользователя и программы должны быть включены в поток.

* Объясните порядок или последовательность, в которой эти элементы должны появляться в этом потоке.

Приведите рисунок, иллюстрирующий состав потока.

V. Требования к планированию. Приведите требования к планированию, которые касаются выполнения задачи, связанной с прикладной системой/программой.

Раздел С: Выполнение задачи

Примечание. Информация, которую следовало бы внести в этот раздел, может содержать, в основном, стандартные процедуры. В таких случаях давайте ссылку на документ, который описывает стандартные процедуры.

I. Начало. Опишите порядок следования данных во входном потоке для того, чтобы выполнить задачу, связанную с прикладной системой/программой.

Дайте первоначальную установку всех переключателей пульта или кнопок.

Укажите, как должна быть выполнена загрузка «шапки» задачи.

Укажите, как фактически начинается выполнение задачи.

Объясните, какие сообщения могут ожидать в течение этого периода.

Отсылайте оператора к приложению «Сообщения оператору» для объяснения каждого сообщения и ответного действия, которое он, оператор, должен принять при каждом сообщении.

II. Выполнение. Опишите все действия, которые должен выполнить оператор в течение выполнения задачи.

- * Перечислите все события, даже очевидные для оператора, которые происходят во время работы системы/программы.

- * Укажите, какие сообщения делаются в течение этого периода, и отсылайте оператора к приложению «Сообщения оператору» для объяснения каждого сообщения.

III. Прерывание. Опишите, когда и при каких условиях оператор может прервать выполнение работы.

- * Дайте последовательность действий, которой он должен придерживаться, чтобы сделать это.

Объясните использование всех вспомогательных программ, таких, как программа разгрузки, которые могут выполняться после того, как было произведено прерывание.

Установите, как оператор может осуществить повторное выполнение задачи, если он вообще это может сделать после того, как она была прервана.

Укажите, какие вспомогательные программы, если таковые имеются, могут быть использованы, чтобы облегчить повторный запуск программы.

IV. Восстановление. Опишите процедуры, если таковые имеются, которым должен следовать оператор, когда задача заикливается или сталкивается с ошибочным условием.

- * Укажите, исходя из чего оператор может сделать вывод, что задача зациклена или столкнулась с ошибочным условием.

V. Завершение.

а. Н о р м а л ь н о е. Перечислите действия, которые оператор должен выполнить, чтобы закончить задачу, которая выполнялась до конца.

Если задача нормально заканчивается сама по себе и никакого вмешательства оператора не требуется, укажите на это.

б. Н е н о р м а л ь н о е. Перечислите действия, которые оператор выполняет, чтобы завершить задачу, вышедшую за пределы отведенных ей ресурсов.

Если оператор ничего не может сделать, скажите об этом.

Раздел D: Выходные данные

I. Данные пользователя. Опишите каждый тип или категорию выходных данных, которые генерируются системой или программой, и в каком виде они могут выдаваться пользователю.

Иллюстрируйте формат каждого типа.

Перечислите те процедуры, которым должен следовать персонал по обработке данных для проверки правильности полученных выходных данных.

II. Организация данных. Опишите организацию памяти для выходных данных, таких, как главные файлы, которые создаются в процессе работы и не могут быть выданы пользователю.

Приложение: Сообщения оператору

Перечислите каждое сообщение, которое оператор может получить в процессе выполнения задачи. Для этого используйте следующую форму.

Сообщение	Объяснение	Ответные действия
Приведите текст сообщения	Укажите, что означает сообщение, и условия, при которых оно выдается	Опишите, какие действия оператор должен выполнить в ответ на сообщение

Модель 4

ПОДГОТОВКА ОПИСАНИЯ ВНУТРЕННЕЙ ЛОГИКИ

Часть I

Предисловие (общий раздел) ().

Рекомендации по эффективному использованию этого описания ().

Оглавление (общий раздел) ().

Введение ().

Глава I. (Наименование первой программы) ().

Раздел А: Логика программы ().

Раздел В: Логика блока программы/подпрограммы ().

I. (Название блока программы/подпрограммы) ().

а. Назначение.

б. Входы.

с. Выходные точки.

д. Используемые таблицы.

е. Выполняемая работа.

} Повторите для каждого блока программы/подпрограммы.

Раздел С: Распечатка программы ().

Приложение: Эксплуатация (название программы) ().

Словарь терминов (общий раздел) ().

Глава I может быть тривиальной в тех случаях, когда описывается только одна программа. В таких случаях заголовок «Глава I» и т. д. не используется. Тогда, когда в инструкции рассматривается ряд программ, которые составляют систему, разделы А, В, С и приложение повторяются для каждой программы, образуя вторую и т. д. главы.

Часть II

Рекомендации по эффективному использованию этого описания

Описание внутренней логики для диалоговых и недиалоговых инженерных, коммерческих прикладных систем служит нуждам программистов, работающих с этой системой, и способствует повышению их квалификации. «Рекомендации по эффективному использованию данного описания» должны указывать на то, что этот документ необходимо использовать в

соответствии с его структурой, т. е. по мере увеличения уровней детализации.

Читатель должен быть введен в курс наилучших способов использования первых разделов:

- *- Логике программы.

- * Логике блока программы/подпрограммы.

Это нужно для того, чтобы он получил представление о том, где найти тот раздел программы, который он должен модифицировать, изменить или модернизировать.

Затем он подготавливается к успешному использованию

- * Листинга программы.

Отсылайте читателя к приложению «Эксплуатация». Используйте для этих рекомендаций форму последовательного изложения или блок-схемы.

Введение

Если описание внутренней логики, которое вы готовите, имеет дело с последовательностью или системой связанных программ, начинайте описание с общего рассмотрения структуры системы в целом, назовите программы, которые она содержит, функций каждой и взаимосвязи этих программ. Включите в инструкцию блок-схемы, показывающие структуру системы. Затем используйте эту же модель для документирования каждой программы в системе, разрабатывая описание каждой программы как отдельный элемент или часть окончательного документа.

Если описание, которое вы готовите, имеет дело с единственной программой, дайте общую информацию по ней во «Введении».

В этом случае заголовок «Глава» в вашем документе использовать не нужно. Ссылайтесь на подпрограммы в разделе В «Логика блока программы/подпрограммы».

Глава I. (Наименование первой программы)

Раздел А: Логика программы

Перечислите здесь главные или все функции или процедуры, выполняемые программой. Связывайте каждую функцию с блоком программы/подпрограмм.

мы, которая предназначена для выполнения этой функции.

Дайте общую блок-схему, которая показывает все функции и возможности этой программы.

- * Отведите один блок для каждой функции.
- * Снабдите каждый блок названием.
- * Поместите эту блок-схему в конце данного раздела инструкции.

Дайте схему, которая показывает все входы в программу и откуда они идут и все выходы и их размещение.

- * Включите пояснительное описание этой схемы.
- * Поместите эту схему после функциональной блок-схемы, рассматриваемой выше.

Отсылайте читателя к разделу В «Логика блока программы/подпрограммы», предназначенному для описания каждого блока программы/подпрограммы, который включается в эту программу.

Раздел В: Логика блока программы/подпрограммы

Опишите здесь каждый блок, содержащийся в программе.

Организуйте эти описания в том порядке, в котором блоки программы и/или подпрограммы появляются в распечатке программы.

Укажите в вашем «Введении», что эти описания:

- 1) общие по природе;
- 2) представляются в наиболее естественной последовательности, т. е. в порядке, в котором блок программы/подпрограммы появляется в распечатке программы;
- 3) каждое дополняется блок-схемой и
- 4) разрабатывается, чтобы облегчить использование программной распечатки блока программы/подпрограммы.

В описании каждого блока программы/подпрограммы пользуйтесь следующей формой.

1. (Название блока программы/подпрограммы)

а. Назначение.

- * Опишите функцию или функции, выполняемые блоком программы/подпрограммы.

- * Перечислите все другие блоки программы/подпрограммы, которые она использует.

в. Входы.

- * Перечислите все точки, которые являются входами в блок программы/подпрограммы.
- * Укажите блок программы/подпрограммы, из которой управление передается в такие точки.
- * Объясните условия, при которых управление передается на каждый вход.
- * Укажите, передается ли управление на каждый вход только один раз или несколько раз в течение выполнения программы.
- * Опишите метод, который используется, чтобы задать любые входные параметры в блоке программы/подпрограммы; опишите области (поля) входных данных.

Если входные данные меняются в соответствии с используемым входом и/или блоком программы/подпрограммы, из которой это управление передается, опишите каждое такое изменение.

с. Выходные точки.

- * Перечислите все точки в блоке программы, из которых управление передается на другой блок.
- * Укажите, на какой блок программы/подпрограммы управление передается из каждой выходной точки.
- * Опишите условия, при которых передается управление.
- * Установите адреса ячеек оперативной памяти, которые используются блоком программы.

Если вывод выходных данных меняется в соответствии с выходными точками и/или блоком программы/подпрограммы, на которые передается управление, опишите каждое такое изменение.

д. Используемые таблицы.

- * Перечислите все таблицы, используемые блоком программы/подпрограммы.
- * Опишите имя и размер каждой.
- * Укажите, генерирует ли фактически этот блок программы/подпрограммы таблицу или таблица строится предварительно некоторым другим блоком программы/подпрограммы.
- * Объясните, зачем используется каждая таблица.

- * Укажите кратко функции каждой таблицы.
- * Опишите все изменения, которые этот блок программы/подпрограммы делает в таблице.

Если несколько блоков программы/подпрограммы используют одну и ту же таблицу, то ее описание не следует повторять каждый раз, а надо просто сослаться на уже имеющееся описание.

В случае, когда эта программа использует много сложных таблиц, вы должны рассмотреть их в заключение этого раздела.

е. Выполняемая работа.

- * Опишите, в общих словах, работу, выполняемую этим блоком программы/подпрограммы.
- * Отсылайте читателя к блок-схеме этого блока программы/подпрограммы.
- * Сгруппируйте все блок-схемы блоков программы/подпрограммы вместе в конце этого раздела.
- * Расположите блок-схемы в алфавитном порядке в соответствии с именами блоков программы/подпрограммы.
- * Приводите функциональную блок-схему, а не блок-схему, каждый блок которой объединяет в себе просто несколько расположенных друг за другом команд.
- * Делайте отметки в распечатке программы там, где читатель будет находить команды, которые фактически выполняют соответствующую этому блоку программы/подпрограммы обработку.
- * Если включается математическая обработка, опишите алгоритм, математическую модель или теорию, на которой основана работа блока программы/подпрограммы. Если необходимо, дайте библиографическую ссылку.

Раздел С: Распечатка программы

- * Приведите распечатку программы. Убедитесь в том, что вами приводится самая последняя распечатка.
- * Если в эту распечатку вносились изменения, проверьте, отражены ли они в инструкции.

Приложение: Эксплуатация (название программы)

Назначение этого приложения заключается в том, чтобы дать программисту-эксплуатационнику специальные инструкции по эксплуатации (например, как добавить новый фактор в таблицу), а также снабдить его описанием специальных технических приемов, используемых в кодировании программы — короче говоря, дать полную информацию о программе, которая поможет программисту-эксплуатационнику модернизировать и изменять эту программу.

Подчеркните такие соображения, как:

- * условия, наложенные размером, типом данных и т. д.;

- * взаимодействие с другими частями системы.

Если требуется, перечислите здесь все соображения, относящиеся к обработке, которые, по вашему мнению, необходимо знать программисту-эксплуатационнику.

Модель 5

ПОДГОТОВКА РУКОВОДСТВА ПОЛЬЗОВАТЕЛЮ ПО ДИАЛОВОЙ ПРИКЛАДНОЙ ПРОГРАММЕ

Часть I

Предисловие (общий раздел) ().

Рекомендации по эффективному использованию этого руководства ().

Оглавление (общий раздел) ().

Раздел А: Введение ().

- I. Назначение ().

- II. Выполняемые функции ().

- III. Ограничения ().

- IV. Дополнительная информация ().

Раздел В: Общая информация ().

- I. Допустимые элементы входных данных ().

- II. Имеющиеся в наличии виды обработки ().

- III. Ограничения ().

- IV. Выходные данные ().

- V. Управляющие директивы диалоговой операционной системы ().

- VI. Управляющие директивы для оборудования/программы ().
- VII. Организация входного потока ().
- Раздел С: Выполнение задания ().
- I. Начало диалоговых операций ().
- II. Загрузка программы (название диалоговой программы) ().
- III. Запуск программы (название диалоговой программы) ().
- IV. Ввод входных данных ().
- V. Процедура обработки ошибок ().
 - a. (Название диалоговой программы) ().
 - b. (Название операционной системы) ().
- VI. Изменение входных данных ().
- VII. Прерывание программы (название диалоговой программы) ().
- VIII. Повторный запуск программы (название диалоговой программы) ().
- IX. Получение выходных данных ().
- X. Завершение работы (название диалоговой программы) ().
- XI. Окончание диалоговых операций ().
- Приложение: Сообщения системы ().
- Словарь терминов (общий раздел) ().

Часть II

Рекомендации по эффективному использованию этого руководства

При подготовке этого раздела «Руководства» включите следующую информацию:

- * пользователь относится к тем, кто связан с вычислительной машиной или пользуется программой коммерческих/технических применений посредством удаленных терминалов.

«Рекомендации», которые вы готовите, должны подробно объяснять:

- * как использовать «Руководство», чтобы сделать наиболее эффективной работу определенной прикладной программы;

- * как использовать ваши инструкции в качестве практического руководства при выполнении программы с удаленного терминала.

Ваша инструкция имеет две ориентации: она эффективно соединяет в себе руководство пользователю и инструкцию для работы, обычно подготовленную для недиалоговых прикладных программ. «Рекомендации по эффективному использованию» должны дать точные указания, как использовать ваше «Руководство» в любой сложившейся ситуации. Рекомендуется ряд последовательных процедур для каждого из этих двух типов использования.

Примечание. Эта модель является руководством при составлении документации по диалоговым программам, написанным для работы в системах разделения времени.

Раздел А: Введение

Приведите всю общую информацию, которую должен знать покупатель для того, чтобы понять, что представляет собой диалоговая прикладная программа и что она способна делать.

- * Пишите этот раздел так, чтобы он был понятен и руководителю и пользователю, который будет использовать эту программу.
- * Составляйте описание, как будто большинство пользователей не знакомо с данной программой или с любой подобной диалоговой прикладной программой.
- * Выберите ту информацию из технического задания, которая определяет необходимость программы и ее особенности.

Короче, не фантазируйте. Начинайте с того, что может показаться наиболее простым, и ведите читателя шаг за шагом к более сложной информации, которую он должен усвоить для того, чтобы понимать программу досконально.

- * Чтобы удовлетворить знающего пользователя, сообщите, что он может опустить этот раздел, если он чувствует, что уже понимает материал, представленный в нем.

I. Назначение. Определите четко и кратко объясните назначение программы.

II. Выполняемые функции. Перечислите и кратко объясните все функции или действия, которые программа выполняет автоматически либо по выбору пользователя.

Будьте точны при указании, какие функции выполняются автоматически, а какие по выбору.

Не вдавайтесь здесь в детали. Эта информация должна быть рассмотрена более подробно в разделе В под заголовком «Имеющиеся в наличии виды обработки».

III. Ограничения. Исходите здесь из того, что существуют ограничения на программу, которые могли бы сделать ее непригодной для определенных нужд пользователя.

* Например, здесь нужно отметить размеры масивов, которые программа может обрабатывать, если они являются ограничивающим фактором.

IV. Дополнительная информация. Здесь включите такую информацию, как: дата последних изменений, какие подпрограммы эта система/программа вызывает, приблизительное время работы программы.

Раздел В: Общая информация

I. Допустимые элементы входных данных. Опишите каждый тип входных данных (обычно именуемый записью), который прикладная программа может обрабатывать.

* Опишите назначение каждого типа записи входных данных.

* Включите иллюстрацию, которая показывает основной формат каждого типа.

* Покажите фактический пример записей для каждого типа.

* Объясните назначение, размер и формат всех полей, содержащихся в отдельных записях, относящихся к каждой категории.

* Отметьте длину записей (в количестве символов или в других единицах), относящихся к каждому типу.

- * Установите, могут ли записи, относящиеся к каждому типу, содержать двоично-десятичные, двоичные, восьмеричные, шестнадцатеричные или другие данные.
- * Установите, являются ли записи, относящиеся к каждому типу, групповыми или негрупповыми, и если групповыми, дайте формат и размер каждой группы.
- * Объясните размер, формат, начальные и конечные записи, запоминающую среду и т. д. файлов и массивов данных (если они есть), в которые организуются записи, связанные с каждой категорией.

II. Имеющиеся в наличии виды обработки. Подробно опишите, какую работу диалоговая прикладная программа способна выполнить автоматически, а какую — по выбору пользователя над каждым из различных типов входных записей, которые пользователь может представить для обработки.

- * Направьте изложение на описание допустимых входных данных, рассмотренных под заголовком «Допустимые элементы входных данных», т. е. укажите, как каждый шаг работы изменяет входные данные и что в результате получается. Если обработка изменяется в зависимости от типа записи входных данных, укажите на это и объясните, как она изменяется.
- * Если последовательность, в которой идет обработка, существенна, нарисуйте и объясните в поясняющем тексте блок-схему, которая показывает точную последовательность, в которой представлен каждый шаг обработки.
- * Не рассматривайте, как программа ведет обработку. Эта информация относится к «Руководству по внутренней логике». Точно укажите, как пользователь отмечает, должна ли программа выполнять необязательную, но имеющуюся в наличии работу.
- * Если возможность выбора обеспечена в форме директив, управляющих программой, рассмотрите цель, формат и возможные параметры каждой директивы.

* Если последовательность работы в программе изменяется в соответствии с выбором вариантов пользователем, установите, как это происходит: укажите, как изменяется работа при выборе каждого варианта, и начертите блок-схему, иллюстрирующую каждое возможное изменение.

* Заметьте, что эта информация является дополнением к информации, приведенной в разделе А под заголовком «Выполняемые функции».

III. Ограничения. Перечислите здесь все ограничения, налагаемые способом работы, используемым системой.

IV. Выходные данные. Опишите типы записей выходных данных, которые выдает эта программа.

Делая это, охватите те вопросы, которые рассмотрены ранее в этой модели под заголовком «Допустимые элементы входных данных».

Не рассматривайте сообщения, генерируемые программой в это время. Они должны быть рассмотрены в приложении «Сообщения системы».

* Отметьте, что такие сообщения выдаются, и отошлите читателя к приложению.

V. Управляющие директивы диалоговой операционной системы. Перечислите управляющие директивы, которые пользователь может вводить для того, чтобы общаться с операционной системой, под надзором которой его программа работает.

Примечание. Если программа использует стандартный набор директив, обеспечьте ссылку на соответствующий документ, который их описывает. Если программа использует нестандартный набор директив, определите его следующим образом:

* укажите имя и мнемонику каждой такой директивы;

* объясните функцию каждой;

* рассмотрите каждую директиву в той последовательности, в которой она должна появляться во входном потоке;

* проиллюстрируйте, как должна выглядеть каждая директива, когда она подготовлена;

- * опишите функцию каждого операнда или параметра, который включается в каждую директиву;
- * если какие-либо директивы, параметры директив или операнды не являются обязательными, укажите, в каких случаях какое их использование выгодно пользователю, и укажите все ограничения на их использование.

З а м е ч а н и е. Если этот материал представлен в инструкции изготовителя операционной системы, просто отошлите пользователя к этой инструкции.

VI. Управляющие директивы для оборудования/программы. Опишите все управляющие директивы, требующиеся:

для оборудования, на котором задание должно выполняться;

для диалоговой программы.

- * Например, управляющая директива ВРЕМЯ *) могла бы потребоваться оборудованию, хотя выбор возможного пути обработки мог бы осуществиться посредством использования специальной управляющей директивы для самой программы.

Ссылайтесь на п. V «Управляющие директивы диалоговой операционной системы» в тех случаях, когда описанные в п. V директивы совпадают с директивами для оборудования в этом разделе.

VII. Организация входного потока. Опишите входной поток, включая управляющие директивы задания, содержащего в себе эту диалоговую программу.

Включите иллюстрацию, которая показывает формат и составляющие потока, а также происхождение каждого элемента, включенного в него.

Раздел С: Выполнение задания

Укажите в начале этого раздела, что:

- * форматы и подробные объяснения всех управляющих директив операционной системы или программы содержатся в разделе В;
- * объяснения и ответы (если они вообще существуют) на все сообщения, генерируемые систе-

*) ВРЕМЯ — директива для оборудования. (Прим. перев.)

мой или программой, содержатся в приложении в конце инструкции.

З а м е ч а н и е. Если какая-либо из процедур, описываемых в следующих параграфах, является стандартной, одинаковой для всех программ, просто отошлите читателя к соответствующему документу.

I. Начало диалоговых операций. Опишите последовательность действий, которой пользователь должен следовать для того, чтобы привести в готовность свой удаленный терминал.

Отметьте, какие управляющие директивы пользователь должен вводить с терминала для того, чтобы начать диалоговые операции.

Перечислите сообщения, которые могут посылаться на терминал пользователя в это время операционной системой, которая управляет диалоговыми операциями.

II. Загрузка программы (название диалоговой программы). Укажите, какие управляющие директивы пользователь должен вводить с терминала для того, чтобы осуществить загрузку диалоговой прикладной программы.

Перечислите все сообщения, выдаваемые операционной системой в это время.

III. Запуск программы (название операционной системы). Укажите, какие управляющие директивы пользователь должен ввести для того, чтобы начать выполнение программы.

* Если существуют различные способы начала работы, опишите директивы, требуемые для каждого.

Перечислите все сообщения, выдаваемые операционной системой или программой в это время.

* Если существуют различные способы начала выполнения, то постарайтесь описать, какие сообщения выдаются при каждом способе.

IV. Ввод входных данных. Опишите последовательность действий, которой пользователь должен следовать при вводе входных данных, которые он хочет обработать.

Перечислите все сообщения, генерируемые ОС или программой в это время.

V. Процедура обработки ошибок.

а. (название диалоговой программы). Объясните, какие ошибки эта программа способна обнаруживать.

* Перечислите ошибки, которые будут приостанавливать обработку входных данных пользователя.

Рассмотрите, что программа делает в случае каждой из этих ошибок.

Перечислите все сообщения, которые программа генерирует при встрече с такой ошибкой.

Расскажите пользователю, что он может сделать, чтобы продолжить работы (если это возможно) после каждой такой ошибки.

б. (название диалоговой программы). Отметьте, какие ошибки операционная система может обнаружить при выполнении программы.

* Перечислите ошибки, которые будут приостанавливать выполнение программы.

* Укажите, что операционная система делает при обнаружении каждой ошибки.

Перечислите все сообщения, генерируемые системой при обнаружении таких ошибок.

Укажите, что пользователь может предпринять для продолжения работы (если это возможно) после каждой ошибки.

Примечание. Если эта информация уже представлена в инструкции изготовителя операционной системы, просто отошлите читателя к этой инструкции.

VI. Изменение входных данных. Опишите, когда и как пользователь может изменить часть входных данных без того, чтобы повторно вводить все свои входные данные сначала.

Отметьте все ограничения операционной системы или программы, налагаемые на внесение любых таких изменений.

Перечислите все сообщения, генерируемые операционной системой или программой при этом.

VII. Прерывание программы (название диалоговой программы). Опишите, когда и как пользователь может прервать выполнение программы. Перечислите сообщения, генерируемые при этом операционной системой.

VIII. Повторный запуск программы (название диалоговой программы). Обсудите, как и при каких условиях программу можно запустить повторно.

- * Если согласование по времени и последовательность действий зависят от того, следует ли повторный запуск за прерыванием, пришедшим от пользователя, за нормальным или за аварийным окончанием программы, постарайтесь описать каждый случай отдельно.

Перечислите все сообщения, генерируемые операционной системой или прикладной программой при выполнении повторного запуска.

IX. Получение выходных данных. Опишите последовательность действий, которой следует пользоваться для получения выходных данных.

- * Если выходные данные получаются автоматически, т. е. выводятся программой без участия пользователя, то укажите, что это так.

Опишите формат, в котором пользователь получает каждый тип выходных данных.

- * Дайте иллюстрацию этого формата.

- * Приведите пример каждого типа выходных данных.

Дайте всю информацию, которая позволит пользователю правильно истолковывать выходные данные.

X. Завершение работы (название диалоговой программы). Установите, как пользователь заканчивает выполнение диалоговой прикладной программы.

Перечислите все сообщения, генерируемые при этом операционной системой или прикладной программой.

XI. Окончание диалоговых операций. Отметьте, какие действия предпринимает пользователь, чтобы закончить диалоговые операции.

Перечислите все сообщения, которые выдаются в это время операционной системой.

Приложение: Сообщения системы

Перечислите здесь каждое сообщение, которое пользователь может получать на своем терминале

или консольной пишущей машинке от операционной системы и от программы.

Опишите сообщения операционной системы и сообщения программы отдельно согласно номеру сообщений или, если они не обозначены номерами, в алфавитном порядке.

Используйте следующую форму для описания каждого сообщения.

Сообщение	Объяснение	Ответные действия
Приведите текст сообщения	Укажите, что сообщение означает, и условия, при которых оно появляется	Опишите, какие действия пользователь должен выполнить в ответ на сообщение

Примечание. Если сообщения операционной системы описаны в инструкции ее изготовителя, то просто отошлите читателя к этой инструкции.

Модель 6

ПОДГОТОВКА РУКОВОДСТВА ПОЛЬЗОВАТЕЛЮ ПО ПОДПРОГРАММЕ

Часть I

Предисловие (общий раздел) ().

Рекомендации по эффективному использованию этого руководства ().

Оглавление (общий раздел) (·).

Раздел А: Введение ().

I. Назначение ().

II. Математический метод или выполняемая функция ().

III. Работа ().

a. Точки входа ().

b. Точки выхода ().

c. Вызываемые подпрограммы ().

d. Ограничения ().

e. Точность ().

- IV. Требования к памяти ().
- V. Временные соотношения ().
- VI. Размещение ().
- Раздел В: Использование ().
- I. Процедура обращения ().
- II. Действия до вызова ().
- III. Дополнительный ввод-вывод ().
- IV. Действия после вызова ().
- Библиография ().
- Словарь терминов (общий раздел) ().

Часть II

Рекомендации по эффективному использованию этого руководства

Этот раздел «Руководства пользователю» для подпрограммы должен указать:

- * как использовать это «Руководство», чтобы ответить на все вопросы по использованию подпрограммы;
- * как использовать это «Руководство» в качестве справочника для получения специфической информации.

Отсылайте читателя к библиографии.

Покажите разработку уровней информации от общего к частному в разделе С.

Раздел А: Введение

I. Назначение. Опишите, для чего подпрограмма была написана.

Укажите язык программирования, на котором написана подпрограмма.

- * Если подпрограмма переписана с существующей подпрограммы, укажите, что это так, назовите уже существующую подпрограмму и объясните, почему переписку считали необходимой или желательной.

II. Математический метод или выполняемая функция. Если подпрограмма является инженерной/научной подпрограммой и математические выкладки в под-

программе нетривиальны, объясните алгоритм, математическую модель или теорию, на которой основано действие подпрограммы.

Отошлите читателя (если это необходимо) к библиографии, где перечислены все книги или статьи, которые дадут ему исчерпывающее объяснение или всестороннее понимание математических выкладок. Если математические выкладки тривиальны, приведите их.

III. Работа. Рассмотрите работу подпрограммы, т. е. то, как она реализует функцию, для выполнения которой она была написана.

Включите алфавитный список всех символических имен и математических символов, использованных в подпрограмме.

* Опишите значение и использование каждого такого символа или имени.

Подкрепите описание работы подпрограммы блок-схемой, которая показывает основные логические этапы ее работы.

* Ограничьте блок-схему размерами одной страницы, если это возможно.

Отошлите читателя, который хочет более детально вникнуть в выполнение подпрограммы, к листингу подпрограммы и объясните, где и как получить копии этого листинга.

Обозначайте каждый квадрат в блок-схеме и ссылайтесь на эти обозначения в описании, чтобы читатель мог легко связывать блок-схему и описание.

а. Точки входа. Перечислите все точки, в которых в подпрограмму можно войти.

Объясните условия, при которых управление передается в каждую такую точку.

б. Точки выхода. Перечислите все точки в подпрограмме, из которых управление может передаваться из подпрограммы.

Опишите, при каких условиях управление передается из каждой точки выхода.

Отметьте точки, в которых управление возвращается в подпрограмму.

Объясните, является возвращение «нормальным» или «аварийным». Для каждого аварийного возвращения приведите причины его возникновения и опи-

шите действия, которые можно или должно предпринять в случае аварийной ситуации.

с. *Вызываемые подпрограммы.* Перечислите все другие подпрограммы, которые вызываются исходной подпрограммой. Кратко укажите назначение каждой. Укажите, откуда берется каждая, если они не включены в один пакет с исходной подпрограммой.

d. *Ограничения.* Перечислите все ограничения, которые налагаются используемым математическим методом или выполняемой функцией/подпрограммой.

Установите все ограничения, присущие подпрограмме, такие, как размеры массивов, которые предназначены для обработки.

Укажите все возможные проблемы, которые могли бы возникнуть в результате взаимодействия исходной подпрограммы с вызываемыми программами или подпрограммами, и их разрешения.

e. *Точность.* Укажите степень точности, обеспечиваемую подпрограммой в выполняемом математическом или статистическом вычислении.

IV. Требования к памяти. Укажите, как много требуется оперативной памяти для подпрограммы и данных, с которыми она работает.

Если это переменная величина, установите верхнюю и нижнюю границы и объясните, почему она изменяется.

V. Временные соотношения. Дайте время, требуемое подпрограмме для решения типичной задачи. Если оно изменяется в широких пределах, укажите диапазон и причины этого изменения. Если это возможно, дайте пользователю уравнение для определения времени вычислений.

VI. Размещение. Установите физическое место подпрограммы.

* Например, если она занесена в каталог системной библиотеки, укажите это.

* Если она находится на бобине магнитной ленты в устройстве ленточной библиотеки, укажите это и опишите имя, которое она носит, а также процедуру, которой нужно следовать для ее извлечения.

Раздел В: Использование

1. Процедура обращения. Опишите оператор ФОРТРАНА или КОБОЛА или макрокоманду ассемблера, которыми нужно воспользоваться для вызова подпрограммы.

Дайте иллюстрацию формата оператора или макрокоманды. Объясните цель и назначение каждого элемента, который может быть использован в имени, операции или полях операндов оператора или макрокоманды.

- * Укажите, какие из элементов обязательные, а какие используются по выбору.

Приведите пример полностью подготовленного оператора или макрокоманды.

В случае операнда ФОРТРАНа рассмотрите:

- * Вызывается ли каждый аргумент по имени или по значению.
- * Следуют ли аргументы за названным оператором или располагаются под общим именем.
- * Дайте следующую информацию об аргументах:
 1. Использование и значение.
 2. Характер: является ли он входным аргументом или выходным и т. д.
 3. Размер: если аргумент является массивом, то укажите количество элементов в массиве.
 4. Тип данных: является ли аргумент числом с фиксированной запятой, плавающей запятой, обычным или с двойной точностью.

Опишите все дополнительные свойства, которые имеют место.

II. Действия до вызова. Перечислите операции, которые должны предшествовать вызываемому оператору, например операции для установки регистров, предшествующие вызову.

Объясните смысл каждой подготовительной команды, которую вы указали для вызывающего оператора или макрокоманды.

Если последовательность операций, предшествующая вызову, изменяется в зависимости от точки входа и существует две или больше точек входа, переходите последовательно от одной точки к другой.

III. Дополнительный ввод-вывод. Опишите все возможные вводы, производимые без помощи вызывающего оператора, макрокоманды или списка аргументов.

Опишите также все выводы, производимые подпрограммой способом, отличным от того, каким обычно передаются результаты в заранее определенный регистр.

Укажите назначение, формат, тип, размер и носитель всех таких вводимых или выводимых данных.

IV. Действия после вызова. Отметьте, какие коды должны следовать за вызывающим оператором или макрокомандой.

* Например, команды, требуемые для получения выходных данных подпрограммы с определенного регистра, для восстановления регистров, для получения необязательного или дополнительного вывода и т. д.

Охватите все, что связано с каждой из этих команд.

Если последовательность команд после вызова изменяется в соответствии с точками выхода и существует две или больше точек выхода, переходите в описании последовательно от одной такой точки к другой.

Библиография

Перечислите для инженерной/научной подпрограммы книги и статьи, которые могли бы дать читателю более полное понимание метода, используемого в подпрограмме для решения задачи.

Модель 7

ПОДГОТОВКА ОПИСАНИЯ БИБЛИОТЕКИ ПРОГРАММ

Часть I

Предисловие (общий раздел) ().

Рекомендации по эффективному использованию этого руководства ().

Оглавление (общий раздел) ().

Раздел А: Введение ().

I. Взаимосвязи программ ().

а. Название группы ().

1. Структура ().

Рисунок — образец библиотечной структурной схемы.

2. Зависимости ().

Раздел В: (имя библиотеки). Описания программ ().

Словарь терминов (общий раздел) ().

Часть II

Рекомендации по эффективному использованию этого описания

Руководство пользователю по библиотеке связанных друг с другом программ предназначается для того, чтобы выделить взаимосвязи между различными программами, которые составляют эту библиотеку.

Этот раздел инструкции должен указать читателю:

- * где найти общую информацию, необходимую для понимания функций, реализуемых программами;

- * как связать различные программы посредством зависимостей и взаимных связей.

Ссылайтесь на описания подпрограмм для более точной информации об особых процессах.

Дайте библиографическую ссылку, если использование библиотеки требует дополнительных знаний.

Раздел А: Введение

Этот раздел «Описания библиотеки» должен определять и описывать содержимое библиотеки.

- * Укажите причины разработки библиотеки и дайте основной материал, который дает пользователю возможность лучше понять происхождение используемых методов.

- * Назовите используемую библиотеку.

- * Дайте общее описание структуры библиотеки.

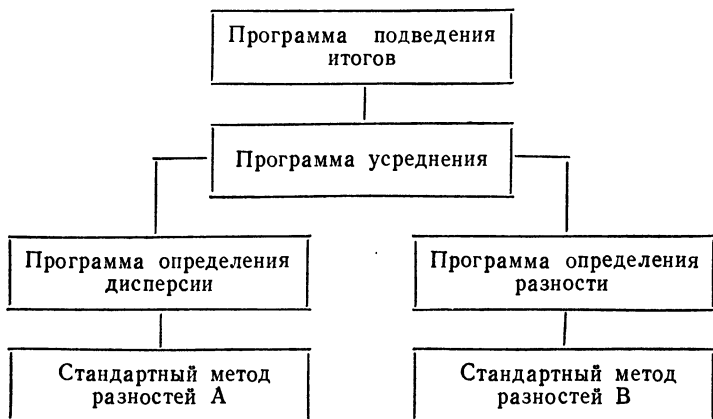
Рассмотрите структуру более подробно в следующем разделе.

1. Взаимосвязи программ. Повторите следующий раздел для каждой основной группы программ. Если библиотека мала, представьте программы в виде одной группы.

а. *Название группы.* Кратко опишите признак, по которому программы объединены в одну группу.

1. *Структура.* Библиотеку или часть библиотеки можно назвать структурой, если существуют программные связи, т. е. программа X не может использоваться без программы Y. Если программных связей не существует, опустите этот подраздел. Используйте рисунок, чтобы показать эти программные связи.

* Например, статистическая библиотека могла бы выглядеть так:



Простая схема зависимостей в библиотеке.

Рисунки должны показывать, какие другие программы требуются каждой программе.

2. *Зависимости.* Перечислите все программы в группе в алфавитном порядке на левой стороне страницы.

Укажите зависимости для каждой программы на правой стороне страницы. Они включают:

* другие требуемые программы,

- * необходимое оборудование (ресурсы),
 - * подготовительные действия
- и т. д.

Раздел В: (имя библиотеки). Описание программ

Опишите программы отдельно. Представляйте их в том же порядке, в котором они приведены в разделе 2.

Если описываемая библиотека является библиотечной подпрограммой, используйте модель 7.

Здесь может быть также использована модель 4.

Модель 8

ПОДГОТОВКА ОПИСАНИЯ ЯЗЫКА АСЕМБЛЕРА

Часть I

Предисловие (общий раздел) ().

Рекомендации по эффективному использованию этого описания ().

Оглавление (общий раздел) ().

Раздел А: Общая информация ().

I. Введение ().

II. Характеристика языка ().

a. Определение ().

b. Используемые символы ().

c. Типы операций ().

III. Аппаратные и программные средства, используемые языком ().

a. Зависимость от аппаратных средств ().

b. Содержание программного обеспечения ().

c. Команды (первый функциональный тип) ().

Раздел В: Операторы языка ().

I. Машинно-ориентированные операторы ().

a. Символические и машинные форматы команд ().

b. Формат представления операторов в языке ().

- II. Расширения машинно-ориентированных операторов ().
 - а. Библиотеки подпрограмм ().
 - III. Команды транслятора (директивные команды) ().
- Раздел С: Использование языка ().
- I. Как писать программу ().
 - а. Пример программы ().
 - II. Как подготовить программу для трансляции ().
 - III. Диагностические сообщения ().
 - а. Диагностика при трансляции ().
 - б. Диагностика при выполнении ().
- Приложения ().
- а. Символы и эквивалентные коды ().
 - б. Указатель команд ().
 - с. Контрольный список программиста ().
 - д. Указатель диагностических сообщений ().
 - е. Таблицы преобразований ().
- Словарь терминов (общий раздел) ().

Часть II

Рекомендации по эффективному использованию этого описания

По языку ассемблера нужно дать несколько описаний, предназначенных для программистов с различной квалификацией. Вы должны:

- * Направить подготовленного программиста к разделам «Введение» и «Словарь специальных терминов».
- * Подчеркнуть важность использования «Оглавления» для поиска разделов, интересующих опытного читателя.
- * Указать на то, что требуется для эффективного использования этого руководства.

Рекомендуются несколько руководств для того, чтобы удовлетворить различные потребности. Во всех случаях должно быть подчеркнуто значение «Резюме» и таблиц. Постоянно ссылайтесь на «Оглавление».

Раздел А: Общая информация

1. Введение. Во «Введении» ориентируйтесь на пользователя, который не знаком с языком ассемблера. Постройте этот раздел как вступительный курс в программирование, объясните основные положения, необходимые для понимания остальной части «Руководства».

II. Характеристики языка.

а. **Определение.** Расшифруйте название языка, если это требуется. Объясните значение каждого сокращения.

* Точно определите, что такое язык ассемблера и как данный язык ассемблера соответствует определению. Кратко опишите, как обрабатываются предложения этого языка.

б. **Используемые символы.** Перечислите символы, используемые в написании операторов языка ассемблера. Дайте набор символов, операторов и т. д., отсылайте пользователя к разделу а «Приложения», к таблицам символов и кодов эквивалентности.

Прокомментируйте правила объединения символов при формировании предложений языка ассемблера.

Укажите типы обрабатываемых ассемблером данных (десятичные, двоичные, восьмеричные и т. д.). Перечислите ограничения.

* Например, здесь можно указать, что ассемблер не признает двоичных символов.

с. **Типы операций.** Определите кратко основные типы операций, простейшие операции, т. е. операции, которые транслируются «один к одному», макрооперации, директивные операции (т. е. те операции, которые предназначены ассемблеру и не транслируются в машинные команды); к таким операциям относятся операции резервирования памяти.

* Объясните причину включения макроопераций и директив в язык и выполняемые ими функции.

Разделите операции по функциональному признаку.

* Рассмотрите, например, в соответствии с этой классификацией операции сдвига, операции, оп-

ределяющие действия с индексными регистрами, операции двоичной арифметики и т. д.

III. Аппаратные и программные средства, используемые языком.

а. Зависимость от аппаратных средств. Объясните в этом разделе, как характеристики схемного обеспечения влияют на адресацию и предложения ввода-вывода.

Опишите связи между символической и физической адресацией, относительной и абсолютной.

Определите различные уровни адресации, такие, как прямая и косвенная.

* Опишите адресацию систем аппаратных средств.

Опишите основные характеристики средств ввода-вывода, последовательный или случайный доступ к файлам. Перечислите все допустимые типы носителей: карты, печатающие устройства, ленты и т. д.

б. Содержание программного обеспечения. Кратко опишите взаимодействие транслятора языка ассемблера с общим программным обеспечением.

Перечислите все ограничения, налагаемые на ассемблер функциями и процедурами операционной системы.

* Например, здесь можно привести такие утверждения, как:

Операционная система занимает 1000 слов памяти

или

Ввод-вывод ассемблера управляется операционной системой.

Далее следует последовательность пунктов, которые составляют описание каждой команды. Эти пункты рекомендуется представлять в порядке, приведенном ниже.

1. Мнемоника операции.

2. Расшифровка мнемоники.

3. Машинный эквивалент мнемоники.

Примечание. Третий пункт не нужен для директивных команд.

4. Описание назначения команды и ее связей с другими командами.

5. Что команда делает в действительности, шаг за шагом, и как она выполняется, если она является командой ассемблера.
6. Время выполнения и память, требуемая для команды.
7. Примеры символической команды с машинными эквивалентами и объяснениями того, что содержится в каждом адресе, регистре и т. д.

Примечание. Целесообразно символические команды и примеры, т. е. пп. 1 и 7, представить так, как они записываются при программировании.

с. Команды (первый функциональный тип). Повторите этот подраздел для каждого типа первичных команд, т. е. сдвиговых команд, команд работы с индекс-регистрами, команд двоичной арифметики и т. д.

Начните раздел с описания типа команд. Определите общие черты и различия между командами, содержащимися в разделе.

* Если различные команды используют одни и те же свойства аппаратной части, упомяните это.

* Объясните связь между ассемблером и системным загрузчиком, если между ними существует взаимодействие.

Отметьте все языковые трансляторы в программном обеспечении, которые используют предложения языка ассемблера либо для промежуточного формата команд, либо для «собственно кодовых» программ.

Не давайте подробной информации, дайте ссылки на соответствующую документацию для дальнейшего изучения.

Раздел В: Операторы языка

1. Машинно-ориентированные операторы. Содержание этого и следующего разделов является стержнем инструкции. Опишите в этом разделе символические команды, которые переводятся в команды машины «один к одному». Будьте очень внимательны при использовании единого формата команд, чтобы пользователь, однажды познакомившийся с форматом, мог легко использовать этот раздел для справок.

Этот раздел включает ряд важных подзаголовков. Они описываются ниже.

а. Символические и машинные форматы команд. Опишите символические и машинные форматы команд. Четко установите между ними связь.

б. Формат представления операторов в языке. Представьте графически формат команд и объясните назначение каждой из его частей так, чтобы читатель хорошо запомнил его до того, как он на практике столкнется с командой.

Включите четкое определение того, как отмечаются необязательные части команд. Приведите пример. Если тип команды близко связан с некоторыми другими типами команд или с особыми свойствами операционной системы, отошлите пользователя к соответствующему документу.

* Например, если существует связь с вводом-выводом операционной системы, то пользователь должен ознакомиться с другой инструкцией для того, чтобы иметь полное представление о вводе-выводе.

II. Расширения машинно-ориентированных операторов. Организация этого раздела подобна организации раздела «Машинно-ориентированные операторы», представленного выше.

Пишущий может объединить два этих раздела в один, но это не рекомендуется. Как и в разделе «Машинно-ориентированные операторы», разделите этот раздел на подразделы.

Первый подраздел должен давать объяснение связей между макрокомандами и командами «один ко одному». Опишите также формат макрокоманд.

* Если свойства адресации для первичных команд и макрокоманд не совпадают, опишите различия.

Второй подраздел должен описывать, как макрокоманды должны быть представлены. Формат должен быть таким же, как формат первичных команд.

* В любом случае представление всех макрокоманд должно иметь одну форму, так чтобы пользователь мог легко обращаться к отдельным макроописаниям.

Третий подраздел должен описывать собственно макрокоманды.

* Если количество макрокоманд велико в языке, разделите их на типы. Повторите третий подраздел для каждого типа макрокоманд.

а. Библиотеки подпрограмм. Укажите, существуют или нет стандартные библиотеки подпрограмм, доступные ассемблеру.

Опишите каждую библиотеку отдельно, используя модель для подготовки описания библиотеки программ. В описании библиотеки представляйте каждую подпрограмму, используя заголовки:

1) Назначение.

2) Действия пользователя (из модели 6).

III. Команды транслятора (директивные команды). Первый подраздел должен подробно объяснять выполняемые директивными командами функции.

Второй подраздел должен описывать, как директивные команды будут представлены в инструкции. Формат их должен быть по возможности таким же, как формат простейших команд и макрокоманд.

* В любом случае представление всех директивных команд должно быть единообразно, так чтобы пользователь мог легко обращаться к отдельным описаниям директив.

Третий подраздел должен описывать собственно директивные инструкции.

* Если директивных команд много в языке, разделите их на типы. Повторите третий подраздел для каждого типа директивных команд.

Раздел С: Использование языка

I. Как писать программу. Изложите пользователю последовательно, что он должен делать, чтобы писать программу на этом языке.

Опишите подробно вид исходной программы. Включите простой пример программирования.

Подробно опишите структуру исходной программы.
а. Пример программы. Дайте пример программы, написанной на этом языке. Постройте программу так, чтобы она использовала как можно больше типов команд и/или возможности, предоставляемые языком.

- * Широко используйте комментарии в программе для объяснения, что делает программа.

По ходу программы описывайте команды и приемы программирования. Сошлитесь на соответствующие разделы «Введения».

II. Как подготовить программу для трансляции. Опишите общую процедуру подготовки программы к трансляции.

На содержание этого раздела влияет: работает ли ассемблер под управлением операционной системы или нет.

- * Если ассемблер работает независимо от операционной системы, опишите все режимы трансляции, форматы управляющих записей, организацию исходной программы и т. д. Представьте описанные пункты графически.
- * Если ассемблер работает под управлением операционной системы, укажите на это и опишите только процедуры ассемблера, не относящиеся к операционной системе. Отошлите читателя к соответствующему разделу модели 11.

Если составляется колода перфокарт, опишите состав колоды. Опишите листинг трансляции. Проиллюстрируйте это графически или приведите примеры.

- * Если язык ассемблера имеет директивные команды, управляющие формой листинга, подробно расскажите о них.

III. Диагностические сообщения. Перечислите все диагностические сообщения трансляции, их смысл и предлагаемые ответные действия на эти сообщения. Не давайте информацию, относящуюся к операционной системе. Вместо этого отошлите читателя к «Руководству пользователю по операционной системе». Представьте диагностические сообщения так, чтобы ими удобно было пользоваться. Приведите их под заголовками:

- а. Диагностика при трансляции.
- б. Диагностика при выполнении.

Далее, разделите основные классы диагностических сообщений на подклассы. Подразделите основной класс а следующим образом:

- 1) Диагностика схемного обеспечения.
- 2) Диагностика распределения памяти.

3) Синтаксическая диагностика.

И т. д.

Подразделите основной класс в следующим образом:

1) Арифметические ошибки.

2) Ошибки формата и подготовки данных.

И т. д.

Отошлите пользователя к «Указателю диагностических сообщений», представленному в «Приложении» этой инструкции.

Приложение

Приведите здесь информацию, которой часто приходится пользоваться и которую удобно иметь в одном месте, а не искать по всей инструкции.

а. Символы и эквивалентные коды. Включите в таблицу все символы, которые существуют в языке или обрабатываются языком. Дайте их коды на картах, ленте печати и при вводе.

б. Указатель команд. Организуйте этот раздел в виде памятки программисту, которую он будет использовать при написании программы.

Перечислите по алфавиту все команды, имеющие mnemonic коды. Дайте расшифровку mnemonic кода, приведите коды машины, используемые регистры.

с. Контрольный список программиста. Представьте, что программист готовит программу к первой трансляции. Представьте, что он решает прежде проверить программу за письменным столом.

Дайте упорядоченный список вопросов, которые не зависят от частного применения языка и в соответствии с которыми программист должен просмотреть свою программу.

Если эти вопросы используются, они должны иметь тот же эффект для предотвращения ошибок, что и предполетный контрольный лист пилота. Они должны помогать программисту в нахождении мелких ошибок как в программе, так и в ее подготовке для трансляции. Этими вопросами могут быть следующие:

1. Правильный ли заголовок?

2. Все ли команды передачи управления имеют соответствующее имя в поле имен? И т. д.

Если этот список подготовлен надлежащим образом, он будет неоценимым помощником в избавлении от мелких, но дорогостоящих ошибок.

д. Указатель диагностических сообщений. Постройте этот раздел так, чтобы программист мог легко использовать его для справок. Перечислите в алфавитном порядке или по номерам все диагностические сообщения трансляции. Дайте их короткое описание и предлагаемые ответные действия. Дайте номер страниц, где каждое диагностическое сообщение рассматривается полностью.

е. Таблицы преобразований. Если машина является двоичной, восьмеричной, шестнадцатеричной и т. д., включите таблицы с десятичными эквивалентами.

Включите сюда описание методов преобразований.

Модель 9

ПОДГОТОВКА РУКОВОДСТВА ПОЛЬЗОВАТЕЛЮ ПО ТРАНСЛЯТОРУ

Часть I

Предисловие (общий раздел) ().

Рекомендации по эффективному использованию этого руководства ().

Оглавление (общий раздел).

Раздел А: Общая информация ().

I. Введение ().

II. Характеристики языка ().

а. Определение языка (название языка) ().

б. Свойства языка (название языка) ().

с. Типы операторов языка (название языка) ().

III. Аппаратные и программные средства, используемые транслятором ().

а. Схемное обеспечение ().

б. Программное обеспечение ().

Раздел В: Структура языка ().

I. Основные операторы языка (название языка) ().

- II. Дополнения к основным операторам языка (название языка) ().
- III. Дополнительные особенности языка (название языка) ().
 - a. Библиотеки подпрограмм ().
 - b. Специальные функции ().
 - c. Использование совмещенного языка ассемблера ().
- Раздел С: Использование транслятора ().
 - I. Как писать программу на языке (название языка) ().
 - a. Пример программы ().
 - II. Как подготовить программу трансляции ().
 - III. Диагностические сообщения ().
- Приложения ().
 - a. Символы языка (название языка) и эквивалентные коды ().
 - b. Указатель операторов ().
 - c. Контрольный список программиста ().
 - d. Указатель диагностических сообщений ().
 - e. Таблицы преобразований ().
 - f. Сравнение языка (название языка) и языка (название предыдущей версии) ().
- Словарь терминов (общий раздел) ().

Часть II

Рекомендации по эффективному использованию этого руководства

Круг работающих с языковым транслятором может включать в себя как лиц, не искушенных в программировании, так и весьма опытных программистов. Этот раздел вашей инструкции должен направлять каждого к разделам, которые ему нужны. Так, вы должны:

- * отослать непрограммиста к вступительным разделам;
- * особо подойти к вводной информации и дать ссылки на другие документы, которые дополняют ваш документ;

- * отослать опытного читателя к описанию различий между этой версией языка и предыдущей (если она была).

Используйте «Оглавление» как путеводитель.

Разработайте отдельные руководства, адресованные пользователям с различным уровнем подготовки.

Раздел А: Общая информация

I. Введение. Адресуйте «Введение» пользователю, который не знаком с данным языком.

Продумайте этот раздел как вступительный курс, объясняющий основные понятия, необходимые для понимания остальной инструкции.

II. Характеристики языка.

а. *Определение языка* (название языка). Объясните причину создания такого языка и типы приложений, для которых он годится. Дайте короткую историю развития языка (если это относится к делу).

- * Отошлите пользователя к разделу «Приложения» для сравнения этой версии с наиболее широко используемой предыдущей версией.

б. *Свойства языка* (название языка). Перечислите символы, используемые данным языком. Приведите набор символов, операторов и т. д.

- * Отошлите читателя к разделу а «Приложений», который содержит таблицы символов и эквивалентных кодов.

Определите элементы этого языка.

- * Например, глаголы, константы, переменные, литералы и т. д.

- * Прокомментируйте правила соединения элементов этого языка в предложения. Объясните пунктуацию при их объединении.

Установите типы данных, обрабатываемых языком, — десятичные, двоичные, буквенно-цифровые и т. д.

- * Например, здесь будет уместно утверждение: *транслятор позволяет использование буквенно-цифровых данных только в операторах формата.*

с. *Типы операторов языка* (название языка). Определите кратко основные типы операторов языка.

- * Например, определите арифметические операторы, управляющие операторы, операторы ввода-вывода и т. д.

Объясните связь между операторами исходной программы, написанной на этом языке и их эквивалентами в рабочей программе.

III. Аппаратные и программные средства, используемые транслятором.

а. Схемное обеспечение. Укажите связь языка с возможными конфигурациями аппаратуры при реализации его на вычислительной машине, если она имеет место.

- * Например, укажите связь операторов ввода-вывода с оборудованием.

Коротко опишите, как транслятор взаимодействует с аппаратурой.

б. Программное обеспечение. Укажите место данного транслятора в средствах обработки данных.

Опишите взаимодействие транслятора с операционной системой и с ассемблером (если такая связь существует).

Перечислите все используемые ограничения, налагаемые на компилятор функциями операционной системы.

* Например, здесь уместно такое утверждение, как: *компилятор ввода-вывода управляется операционной системой.*

Не давайте подробной информации, сделайте ссылки на «Руководство пользователю по операционной системе».

Раздел В: Структура языка

I. Основные операторы языка (название языка). Начните этот раздел с графического представления формата, используемого в каждом операторе.

Представьте каждый оператор в этом формате. Начинайте каждый оператор с новой страницы.

Представьте операторы в алфавитном порядке с учетом главной классификации, рассмотренной во «Введении».

В описании каждого оператора избегайте включения посторонней информации. Таким образом, описание каждого оператора будет независимым.

II. Дополнения к основным операторам языка (название языка). В этом разделе представьте дополнительные операторы языка, если таких нет — опустите этот раздел.

Важно, чтобы дополненные операторы отделялись от основных для того, чтобы ясно указать, какие операторы могут быть использованы для обеспечения междумашинной совместимости.

III. Дополнительные особенности языка (название языка).

Представьте в этом разделе дополнительные возможности, предоставляемые языком.

Например, могут быть использованы следующие подзаголовки.

а. Библиотеки подпрограмм. Опишите каждую библиотеку отдельно, используя модель 7.

* При описании каждой библиотеки опишите каждую подпрограмму, используя заголовки «Назначение» и «Действия пользователя» из модели для подготовки руководства пользователю по подпрограмме.

б. Специальные функции. Опишите отдельно каждую такую функцию, используя заголовки. I. «Назначение» и II. «Действия пользователя» из модели 6.

с. Использование совмещенного языка ассемблера. Покажите, как операторы языка ассемблера можно чередовать с операторами данного языка.

Раздел С: Использование транслятора

I. Как писать программу на языке (название языка). Расскажите пользователю последовательно, что он должен делать, чтобы писать программу на этом языке.

Опишите полностью вид исходной программы. Включите пример программирования с простыми операторами.

Опишите подробно структуру исходной программы.

а. Пример программы. Приведите пример программы, написанной на языке. Постройте программу так, чтобы она использовала как можно больше различных типов операторов и/или приемов программирования, предоставляемых языком.

Широко используйте комментарии в программе для последовательного объяснения того, что программа делает.

По ходу программы описывайте операторы, делайте ссылки на соответствующие разделы «Введения».

II. Как подготовить программу к трансляции. Опишите последовательность действий, необходимую для подготовки программы к трансляции. На содержание этого раздела влияет, работает ли транслятор под управлением операционной системы.

- * Если транслятор работает независимо от операционной системы, опишите полно все режимы компилятора, форматы управляющих записей, организацию исходной программы. Дайте графическое представление.

- * Если компилятор работает под управлением операционной системы, укажите на это и опишите только процедуры транслятора, не относящиеся к операционной системе. Отошлите пользователя к соответствующему разделу «Руководства пользователю по операционной системе».

III. Диагностические сообщения. Перечислите все диагностические сообщения, их смысл и предлагаемые ответные действия.

Не давайте информацию по операционной системе. Вместо этого отошлите пользователя к «Руководству пользователю по операционной системе».

Представьте диагностические сообщения в удобном для использования порядке под заголовками основной классификации.

- * Например, такая классификация может включать: а) диагностику при трансляции, б) диагностику при выполнении.

Затем подразделите основные классы диагностических сообщений на подклассы. Продолжайте, как в нижеприведенном примере, делить основной класс а) следующим образом:

1) Диагностика, относящаяся к аппаратным средствам.

2) Диагностика распределения памяти.

3) Диагностика синтаксиса и т. д.

Подразделите основной класс в следующим образом:

1) Арифметические ошибки.

2) Ошибки в подготовке данных и формате и т. д.

Отошлите пользователя к «Указателю диагностических сообщений», представленному в «Приложениях» этого руководства.

Приложения

Соберите здесь всю информацию, которая часто используется программистом:

а. Символы языка (название языка) и эквивалентные коды. Включите в таблицу все символы, которые существуют в языке или обрабатываются языком соответствующими кодами на перфокартах, лентах, печатающих устройствах и внутренними кодами.

б. Указатель операторов. Организуйте этот раздел в виде памятки программиста, которую он использует при написании программы.

Перечислите в алфавитном порядке все операторы, включая все вариации в формате.

с. Контрольный список программиста. Предположите, что программист готовит программу к первой трансляции. Допустите, что он решает прежде проверить программу за письменным столом.

Дайте пронумерованный список вопросов, которые не зависят от частного применения и в соответствии с которыми он должен просмотреть свою программу.

Если эти вопросы используются, они должны иметь тот же эффект для предотвращения ошибок, что и контрольный предполетный лист пилота. Они должны помогать программисту в нахождении мелких ошибок как в программе, так и в ее подготовке к трансляции.

Задайте такие вопросы, как:

1. Есть ли соответствие между метками в операторах перехода и метками операторов, на которые этот переход осуществляется?

2. Все ли метки правильны? И т. д.

Если этот список подготовлен надлежащим образом, он будет неоценимым помощником в избавлении от мелких, но дорогостоящих ошибок.

d. Указатель диагностических сообщений. Перечислите в алфавитном порядке или по номерам все диагностические сообщения транслятора. Дайте их краткие описания и предполагаемые ответные действия.

Укажите номер страницы, где каждое диагностическое сообщение рассматривается полностью.

e. Таблицы преобразований. Если машина двоичная, восьмеричная, шестнадцатеричная и т. д., включите таблицы с десятичными эквивалентами. Приведите методы преобразования чисел.

f. Сравнение языка (название языка) и языка (название предыдущей версии). Приведите перечисление всех различий между этим языком и предыдущей версией этого языка. Это удобно сделать в виде таблицы.

Модель 10

ПОДГОТОВКА ОПИСАНИЯ ЯЗЫКА ВВОДА-ВЫВОДА

Часть I

Предисловие (общий раздел) ().

Рекомендации по эффективному использованию описания ().

Оглавление (общий раздел) ().

Раздел А: Общая информация ().

I. Введение ().

II. Характеристики языка ().

a. Определение языка (название языка) ().

b. Символы языка (название языка) ().

III. Аппаратные и программные средства, используемые языком.

- а. Устройства ввода-вывода, на которые ориентирован этот язык ().
- б. Содержание программного обеспечения ().

Раздел В: Операторы языка ().

- I. Как определить файл ввода-вывода ().
- II. Операторы ввода-вывода ().
 - а. Форматы операторов ввода-вывода ().
 - б. Формат, используемый для представления каждого оператора ввода-вывода ().
 - с. (Название типа команд) ().
- III. Расширение основных операторов языка (название языка) ().

Раздел С: Использование языка ().

- I. Как пользоваться операторами ().
- II. Как транслировать программу, содержащую операторы ввода-вывода ().
- III. Диагностические сообщения ().

Приложения ().

- а. Символы и эквивалентные коды ().
- б. Указатель команд ().
- с. Контрольный список программиста ().
- д. Указатель диагностических сообщений ().

Словарь терминов (общий раздел) ().

Часть II

Рекомендации по эффективному использованию этого описания

В «Рекомендации по эффективному использованию описания» вы должны включить следующие указания:

- * Где найти описание элементов языка и правила для построения операторов.
- * Как использовать ваше «Описание» для справок в целях лучшего использования возможностей, предоставляемых языком.

Дайте информацию об основополагающем материале, который необходим для понимания этого «Описания».

Раздел А: Общая информация

1. Введение. В этом разделе дайте основные идеи, необходимые для понимания остальной части «Описания».

Ориентируйте «Введение» на пользователя, который не знаком с этим языком ввода-вывода.

Кратко опишите вычислительную машину или ряд вычислительных машин, для которых этот язык ввода-вывода написан.

- * Упомяните такие возможности, как мультипрограммная работа, работа в реальном масштабе времени и т. д.

- * Включите краткое описание формата машинной команды, бита, символа, слова, специальных регистров, устройств ввода-вывода.

II. Характеристики языка. Четко сформулируйте, что такое язык ввода-вывода, и объясните, как данный язык ввода-вывода соответствует этому определению.

Определите, что такое система управления вводом-выводом, и опишите возможности, которые она предоставляет пользователю.

Коротко опишите, как этот язык обрабатывается ассемблером или компилятором для установления связи с системой управления вводом-выводом.

Опишите связь между системой управления вводом-выводом, различными системами программного обеспечения и вычислительной машиной.

а. Определение языка (название языка). Дайте русский эквивалент *) сокращений языка.

Опишите все особенности, которые отличают этот язык от других языков ввода-вывода.

б. Символы языка (название языка). Перечислите символы, используемые при написании команд языка ввода-вывода.

- * Дайте перечень символов, операторов и т. д.

- * Отсылайте читателя к разделу а «Приложения».

Перечислите правила для объединения символов в целях формирования предложений языка ввода-вывода.

*) В оригинале — «английский эквивалент». (Прим. перев.)

Укажите, какие типы данных обрабатываются языком ввода-вывода, т. е. двоично-десятичные, двоичные, алфавитно-цифровые и т. д.

III. Аппаратные и программные средства, используемые языком.

а. Устройства ввода-вывода, на которые ориентирован этот язык. Перечислите и опишите все устройства ввода-вывода, на которые ориентирован этот язык.

* Не уточняйте связи между отдельными командами и оборудованием.

Коротко укажите средства, с помощью которых данные передаются от одного устройства к другому.

Приведите все ограничения, установленные пользователем устройствами ввода-вывода.

* Например, здесь уместно такое утверждение, как: *ленты нельзя читать в обратном направлении.*

Укажите, как оборудование может быть использовано оптимально, если на это можно влиять программно.

Приведите все важные временные соотношения.

* В частности, укажите, как много времени проходит с момента прочтения данных устройствами до их готовности для дальнейшей работы.

Где это необходимо, сделайте ссылки на инструкции, подробно описывающие устройства.

* Если машина имеет оборудование, для которого существует свой особый язык ввода-вывода, отошлите читателя к инструкции по тому языку.

б. Содержание программного обеспечения. Покажите, как язык ввода-вывода взаимодействует с другими элементами программного обеспечения.

Отошлите читающего к подробным инструкциям, содержащим информацию по программному обеспечению, а также к разделу общей информации.

* Если этот язык ввода-вывода должен использоваться с языком ассемблера (или компилятора), работающего под управлением операционной системы, отошлите читателя к «Руководству пользователю по операционной системе».

- * Если язык ввода-вывода может быть использован с более чем одним языком ассемблера, упомяните об этом и отошлите читателя к соответствующим инструкциям.

Коротко опишите взаимодействие операционной системы с программой, содержащей команды ввода-вывода.

Перечислите ограничения, налагаемые на систему управления функциями и процедурами операционной системы.

- * Например, здесь можно сделать такое утверждение, как: *операционная система не разрешает использование системного устройства ввода с перфокарт.*

Не давайте подробной информации. Это уменьшит необходимость в обновлении инструкции, если в операционной системе будут сделаны изменения.

Предложите возможный круг задач и при этом дайте ссылки на другие руководства.

Раздел В: Операторы языка

I. Как определить файл ввода-вывода. Приведите все стандарты и условные обозначения, установленные для описания содержимого файла данных.

Если понятие файла отличается для каждого устройства ввода-вывода, дайте все средства описания содержимого файла.

- * Опишите систему присвоения названий.
- * Графически представьте признаки начала и конца, показывая такие элементы, как контрольные суммы, указатели цепочек (указатели связей) и т. д.

II. Операторы ввода-вывода. Содержание этого и следующего разделов является основой инструкции. Будьте очень внимательны при использовании единого стиля написания при представлении команд, так чтобы пользователь, однажды познакомившись с форматом, мог легко использовать этот раздел для справок. Этот раздел может включать ряд важных подзаголовков. Они приводятся ниже.

а. *Форматы операторов ввода-вывода.* Опишите форматы операторов ввода-вывода.

Представьте эти форматы графически.

б. Формат, используемый для представления каждого оператора ввода-вывода. Опишите, как будет представляться каждый оператор ввода-вывода.

Представьте формат оператора графически и объясните назначение каждой его части, так чтобы читатель имел четкое представление о формате до того, как он столкнется с оператором.

Дайте четкое определение того, как отличают используемые по выбору части операторов от их обязательных частей.

Приведите пример.

Далее предлагаются пункты, которые должны присутствовать в описании каждого оператора. Эти пункты рекомендуется представлять в порядке, данном ниже.

1. Оператор ввода-вывода.
2. Русский эквивалент мнемоники, данной в п. 1.
3. Код языка ассемблера, генерируемый оператором ввода-вывода.
4. Описание назначения оператора и его связей (если они есть) с другими командами.
5. Описание работы оператора.
6. Время выполнения и память, которая требуется для работы оператора.
7. Примеры использования этого оператора.

З а м е ч а н и е. Целесообразно операторы ввода-вывода и примеры представлять в таком виде, какой они имеют при написании программ.

с. (Название типа команд). Повторите этот подраздел для каждой главной категории операторов ввода-вывода, например команд чтения, записи и т. д. Если команд мало, представьте их в одной группе.

III. Расширение основных операторов языка (название языка). В этом разделе перечислите и опишите команды ввода-вывода для обращения к устройствам ввода-вывода. Если таких операторов нет, то опустите этот раздел.

Опишите каждое расширение, используя пункты, перечисленные в подразделе II, б.

Раздел С: Использование языка

I. Как пользоваться операторами. В этом разделе продемонстрируйте, как должны использоваться операторы ввода-вывода для выполнения различных действий по вводу-выводу, которые допускаются в этом языке.

* Сообщите пользователю о том, что, если действие не рассматривалось в этом разделе, его нельзя выполнить.

Перечислите набор команд, которые обеспечивают получение данных перед их использованием.

Внесите в список особый оператор, используемый для передачи данных из среды ввода-вывода в центральную вычислительную машину (или из нее) или в другую среду.

Перечислите особые операторы, необходимые для достижения совмещения.

II. Как транслировать программу, содержащую операторы ввода-вывода. Опишите только те добавочные процедуры/соображения, к которым приводит включение операторов ввода-вывода в программу.

III. Диагностические сообщения. Перечислите все диагностические сообщения ввода-вывода, их смысл и предполагаемые ответные действия.

Не давайте информации, относящейся к операционной системе. Вместо этого отошлите пользователя к «Руководству пользователю по операционной системе».

Представьте диагностические сообщения, пользуясь следующими заголовками:

a. Диагностика при трансляции.

b. Диагностика при выполнении.

Далее подразделите диагностику, даваемую под этими заголовками, на типы. Продолжайте по данному выше примеру деление следующим образом:

1. Диагностика, относящаяся к аппаратным средствам.

2. Синтаксическая диагностика.

Выделите в пункте информации о несоответствиях в подготовке данных и формате и т. д. Отошлите читателя к пункту d «Приложений».

Приложения

Здесь представьте все таблицы и другие типы итоговой информации. Ниже приводится несколько наиболее важных из них.

а. Символы и эквивалентные коды. Включите в таблицу все символы, которые существуют в языке или обрабатываются языком, с их кодировкой на перфокартах и перфоленте, внутренними кодами, кодами печати и т. д.

б. Указатель команд. Сделайте этот раздел памяткой программисту, которую он использует при написании программы.

Перечислите в алфавитном порядке все команды, включая мнемонические коды, русский эквивалент этих кодов и т. д.

с. Контрольный список программиста. Предположите, что программист готовит программу, содержащую команды ввода-вывода, к первой трансляции.

Предположите, что он начинает сначала проверять программу за письменным столом. Дайте пронумерованный список вопросов, не зависящих от частного применения и в соответствии с которыми он должен просмотреть свою программу.

Если эти вопросы используются, то они будут иметь тот же эффект в предотвращении ошибки, что и предполетный контрольный лист пилота.

К этим вопросам можно отнести следующие:

1. Проверьте правильность написания заголовка.
2. Убедитесь в отсутствии цифровых символов в поле операции.
3. Убедитесь, имеют ли все команды ввода-вывода соответствующее количество аргументов, и т. д.

Если этот список подготовлен соответствующим образом, он будет незаменимым помощником в избавлении от мелких, но дорогостоящих ошибок.

д. Указатель диагностических сообщений. Пишите этот раздел так, чтобы программисту было удобно работать с ним.

Перечислите в алфавитном порядке или по порядку номеров все диагностические сообщения ввода-вывода.

Дайте краткие описания их значения и предполагаемые ответные действия. Дайте номера страниц, где каждое диагностическое сообщение рассматривается полностью.

Модель 11

ПОДГОТОВКА РУКОВОДСТВА ПОЛЬЗОВАТЕЛЮ ПО ОПЕРАЦИОННОЙ СИСТЕМЕ

Часть I

Предисловие (общий раздел) ().

Рекомендации по эффективному использованию этого руководства ().

Оглавление (общий раздел) ().

Раздел А: Общая информация ().

I. Введение ().

II. Подход к операционной системе ().

Пакетная обработка заданий.

Гибкость системы.

Управление данными.

III. Основные понятия операционной системы
(название) ().

а. Функции и техническое оснащение ().

Общее представление о работе.

Необходимое оборудование операционной системы.

Пакетная и дистанционная обработка.

б. Преимущества ().

IV. Структура (название системы) ().

Функции.

Системные файлы.

V. Стандарты и условные обозначения ().

Раздел В: (название системы) и программные и аппаратные средства ().

I. Конфигурации оборудования ().

а. Требования к машине ().

Центральный процессор.

Устройства ввода-вывода: на перфокарты, на ленты-диски, на удаленные пульта, на устройства дистанционной связи.

б. Используемое дополнительное оборудование (). (Повторите заголовки, перечисленные выше.)

Программное обеспечение ().

а. Компоненты системы и их назначение ().

1. Мониторы:

1) Управления заданиями.

2) Управления заданием.

3) Для обслуживания удаленных пультов.

2. Загрузчик связей:

3. Трансляторы:

1) Ассемблер.

2) ФОРТРАН.

3) КОБОЛ.

4) PL/I.

5) Генератор отчетов.

4. Библиотеки:

1) Математическая.

2) Статистическая.

5. Служебные программы:

1) Управление данными.

2) Обработка данных.

3) Нарастивание программ.

С: Удобства, предоставляемые системой (название) ().

Режимы работы ().

а. Мультипрограммирование.

б. Мультиобработка.

с. Дистанционная обработка.

Управление вводом-выводом ().

а. Управление файлами.

б. Средства связи.

Вспомогательные средства ().

а. Трансляторы.

б. Библиотеки.

с. Служебные программы.

Средства обнаружения ошибок ().

а. Отладочные пакеты.

б. Генераторы проверочных данных.

Помощь в руководстве ().

а) Учет.

б. Планирование.

Д: Компоненты системы (название) ().

(Первый) монитор.

а. Структура. } Повторите для каждого вида
б. Функции. } мониторов.

- II. Загрузчик связей.
 - а. Функции.
 - III. Трансляторы.
 - а. Уровень.
 - б. Дополнительные особенности.
 - с. Ограничения.
 - IV. Библиотеки.
 - а. Содержание.
 - б. Структура.
 - с. Ограничения.
 - V. Служебные программы.
 - а. (Первая) группа.
 - 1. Структура.
 - 2. Возможности.
 - 3. Ограничения.
- } Повторите для все
служебных пакетов.
- VI. Средства обнаружения ошибок.
 - а. Функции.
 - б. Структура.
 - с. Возможности.
 - д. Ограничения.
- Раздел Е: Взаимодействие программиста с системой ().
- I. Введение к системным связям ().
 - а. Управляющие записи.
 - б. Сообщения ввода (работа системы).
 - II. Язык управления заданиями ().
 - а. Форматы.
 - б. Условные обозначения.
 - с. Ограничения.
 - III. Использование мониторов ().
 - а. Как задавать инструкции монитору.
 - 1. Управляющие записи:
Стандартные. Необязательные.
 - 2. Сообщения:
 - 1) Код/номер, текст.
 - 2) Смысл.
 - 3) Реакция.
 - 3. Ограничения.
 - 4. Состав колоды перфокарт.
 - б. Как обработать пакет.
 - с. Как запросить мультиобработку.
 - д. Как произвести автоматическое планирование.

IV. Использование загрузчиков ().

- a. Как задавать инструкции загрузчику.
- b. Как начать загрузку программы.
- c. Как инициировать поиск файла.
- d. Как вызывать подпрограммы.

Подразделы с 1 по 4 раздела IIIa должны быть повторены в разделе IVa.

Раздел F: Связь с компонентами специального назначения ().

I. Использование трансляторов ().

- a. Как собрать пакет.
 1. Управляющие записи.
Стандартные.
Необязательные.
 2. Сообщения.
 3. Ограничения.
- b. Как скомпилировать/выполнить программу (повторите перечисленные выше подразделы с 1 по 3).
- c. Состав колоды перфокарт ().
- d. Специальные замечания к (название транслятора) ().

Раздел d следует повторить для каждой программы обработки (в системе.)

II. Использование системы отладки ().

- a. Как использовать отладку на исходном языке.
 1. Управляющие записи.
 2. Состав колоды перфокарт.
 3. Ограничения.
- b. Как произвести прикидку времени выполнения.
 1. Управляющие записи.
 2. Состав колоды перфокарт.
 3. Ограничения.

III. Использование библиотек ().

- a. Как вызвать библиотеку.
 1. Управляющие записи.
 2. Состав колоды перфокарт
 3. Ограничения.
- b. Как изменить библиотеку.
 1. Управляющие записи.
 2. Состав колоды перфокарт
 3. Ограничения.

} повторите для каждой библиотечки.

} Повторите для каждой библиотечки.

с. Как подключить дополнительные библиотеки в систему.

1. Управляющие записи.

2. Состав колоды перфокарт.

Ограничения.

IV. Использование служебных программ ().

а. Управляющие записи.

б. Состав колоды перфокарт.

с. Ограничения

(Повторите этот раздел для каждой имеющейся служебной программы.)

Приложение А: Резюме ().

I. Возможности системы.

II. Управляющие записи.

III. Состав оборудования.

IV. Состав программного обеспечения.

V. Ограничения.

Приложение В: Указатели ().

I. Возможности системы.

II. Управляющие записи.

III. Сообщения системы.

IV. Компоненты системы.

V. Рисунки составов колод перфокарт.

Словарь терминов (общий раздел) ().

Часть II

Рекомендации по эффективному использованию этого руководства

Информация, представленная в «Руководстве по операционной системе», предназначена для широкого круга читателей. Она состоит из нескольких уровней детализации и должна удовлетворить потребности всех читателей.

«Рекомендации по эффективному использованию» должны содержать среди других указаний инструкции для использования руководства:

- * руководящим персоналом, как техническим, так и нетехническим, который нуждается в общем обзоре;

- * случайным программистом, который нуждается как в специфической информации, так и в содержательном описании;
- * прикладным программистом, который ищет точную информацию о выполнении специфических задач;

Дайте вкратце полные руководства для всех групп читателей, чтобы гарантировать, что материал, который вами представлен, используется в соответствии с вашими намерениями. Особенно здесь уместен набор действий, выполняемых шаг за шагом и иллюстрирующих, как читатели должны действовать при пользовании этим руководством, чтобы удовлетворить свои потребности.

Раздел А: Общая информация

I. Введение. Введите основные понятия, которые являются фундаментальными для понимания остальной части «Руководства».

Адресуйте «Введение» всем пользователям в соответствии с их подготовкой.

Свободно используйте подзаголовки, чтобы задать структуру вашему материалу и помочь пользователю в нахождении им своего уровня. Организуйте раздел А в соответствии со следующими основными подзаголовками и разверните по мере необходимости каждую тему.

II. Подход к операционной системе. Начните этот раздел с объяснения читателю причин, которые лежат в основе развития операционных систем.

Представьте операционные системы как программное обеспечение.

Кратко опишите назначение операционных систем. Не вдаваясь в детали, объясните такие понятия, как пакетная обработка заданий, гибкость системы, управление данными.

- * Включите некоторые дополнительные случаи обработки. Излагая материал, постоянно имейте в виду операционную систему, не включайте особенности, которые относятся к следующему разделу III.

III. Основные понятия операционной системы (название).

а. **Функции и техническое оснащение.** Опишите специфические функции, влияющие на оснащение рассматриваемой системы.

- * Излагайте этот раздел в форме обсуждения основных особенностей операционной системы. Используйте такие подзаголовки, как «Общее представление о работе системы».

Организируйте материал так, чтобы возможно полнее охарактеризовать рассматриваемую систему.

- * Перечислите, например, необходимое оборудование.

- * Если речь идет об обработке, четко различайте пакетную и дистанционную обработку.

б. **Преимущества.** Опишите преимущества рассматриваемой системы. Придайте вашему тексту двухуровневую структуру, если это уместно.

- * Непосредственное обсуждение внешних преимуществ — повышенная эффективность, гибкость использования и т. д. — для административных работников.

- * Непосредственное обсуждение внутренних преимуществ — легкость программирования, специальные библиотеки, отладочные особенности и т. п. — для программирующего персонала.

Если нужно, используйте подзаголовки.

IV. Структура (название системы). Проиллюстрируйте внутреннюю структуру системы, перечислите модули, которые образуют систему. Покажите, как эти модули связаны между собой, как они взаимодействуют при реализации различных системных функций. Приведите блок-схемы, демонстрирующие их различные комбинации, если такое графическое представление может улучшить понимание. Перечислите системные файлы и их характеристики, такие, как носитель, размещение, устройство, размер и т. д.

V. Стандарты и условные обозначения. Определите все стандартные термины, которые используются в этом «Руководстве».

Объясните все специальные условные обозначения, которые используются в описании рассматриваемой системы.

Раздел В: (название системы) программные и аппаратные средства

Опишите рассматриваемую систему в ее обычной работе.

I. Конфигурация оборудования. Начните с описания основных составных частей машины, необходимых для использования системы.

а. Требования к машине. Идите от наиболее простых требований к оборудованию и связанных с ними функций системы к наиболее сложным конфигурациям. Перечислите те части системы, для которых допустимы отклонения: центральный процессор, печать/перфорация, ленты/диски, удаленные пульта и устройства дистанционной связи.

б. Используемое дополнительное оборудование. Проведите различие между конфигурациями, которые возможны для стандартных применений, допускающих такую гибкость, как взаимозаменяемые носители для определенных файлов, и теми, которые, по существу, являются расширениями. Примером последнего могло бы быть применение удаленных пультов (консолей) как специальных устройств для обеспечения расширенного использования.

* Посвятите подраздел каждой конфигурации, если это возможно. Укажите границы выполняемых изменений, диктуемых расширениями.

* Опишите отличия в работе различных допустимых конфигураций.

Для большей ясности при представлении этого материала рекомендуется использовать табличную форму везде, где это допустимо.

II. Программное обеспечение. Представьте как единое целое все программное обеспечение, используемое в рассматриваемой системе.

а. Компоненты системы и их назначение. Установите связь операционной системы с функционированием и использованием других частей программного обеспечения.

Попытайтесь сориентировать читателей на программное обеспечение как на единое целое и одну из составных частей системы полной обработки информации.

Опишите рабочее назначение различных категорий программного обеспечения. Если основной каркас разработан, опишите части программного обеспечения в каждой категории. Используйте подзаголовки, чтобы указать категории.

* Например, мониторы, трансляторы, служебные программы.

Разделяйте различные составляющие внутри каждой группы. Для приведенного выше примера могло бы появиться следующее описание:

1. М о н и т о р ы:

- 1) Для управления заданиями.
- 2) Для управления заданием.
- 3) Для обслуживания удаленных пультов.

2. З а г р у з ч и к с в я з е й.

3. Т р а н с л я т о р ы:

- 1) Ассемблер.
- 2) ФОРТРАН.
- 3) КОБОЛ.
- 4) PL/1.
- 4) Генератор отчетов.

4. Б и б л и о т е к и:

- 1) Математическая.
- 2) Статистическая.

5. С л у ж е б н ы е п р о г р а м м ы:

- 1) Управление данными.
- 2) Преобразование данных.
- 3) Нарращивание программ.

Проиллюстрируйте взаимодействие составных частей, приведя блок-схемы. (Блок-схемы, приведенные в этом разделе, отличаются по ориентации и деталям от тех, которые могли бы быть подготовлены для раздела, описывающего структуру. Блок-схемы, приведенные здесь, показывают действительную связь и взаимодействие составных частей, а не функциональный состав.)

*Раздел С: Удобства, предоставляемые
системой (название)*

Перечислите все возможности системы, имеющие интерес для программиста. Организуйте и сгруппируйте информацию в этом разделе в соответствии с

характеристиками и составом рассматриваемой системы.

Будьте кратки в этом разделе. Используйте списки, таблицы, диаграммы везде, где они применимы.

Попытайтесь представить рассматриваемую систему в ряде аспектов — как помощника при отладке, для преобразования данных и т. д.

* Каждую из возможностей, которыми обладает система, снабдите руководством по использованию.

Для операционной системы этот раздел лучше всего построить в соответствии с ходом работы. Так, например, предлагаемые разделы могли бы быть такими, как описано ниже.

I. Режимы работы. Здесь описываются такие возможности, как мультипрограммирование, мультиобработка и дистанционная обработка. Должны быть также перечислены возможности контроля и повторного пуска программ.

II. Управление вводом-выводом. Здесь может быть приведена информация, касающаяся управления файлами, смены носителей, средств связи и т. д.

III. Вспомогательные средства. Материал, представленный здесь, мог бы быть подразделен по функциональным уровням для того, чтобы учесть нужды расширенной системы.

Подразделы или, где необходимо, независимые разделы описывают те удобства, которые созданы для того, чтобы помочь программисту вникнуть в описываемую систему при проведении обработки типичных задач.

* Например, трансляторы и библиотеки перечислены по названиям. Группы служебных программ показаны по категориям, если такая классификация применяется.

IV. Средства обнаружения ошибок. Опишите возможности отладочного пакета, особенности специального вывода на печать, контрольные предложения исходного языка, генераторы проверочных данных и т. п.

V. Помощь в руководстве. Перечислите системное обеспечение для автоматического учета, планирования, регистрации или других организационных работ.

Используйте общую терминологию, применяемую в системном программировании, избегайте деталей, которые, по существу, можно отнести к разделу Е.

Раздел D: Компоненты системы (название)

Расположите компоненты системы в соответствии с функциональной классификацией.

Представьте специфическую информацию о программах, которые входят в рассматриваемую систему.

Проведите различие между похожими, но различными программами, например программами, которые выполняют некоторые похожие функции, но в разное время, с различными данными и т. п. Излагайте информацию в соответствии с классификацией программ.

- * Например, опишите структуру и функции мониторов.

Расположите материал под заголовками: «Структура», «Функции».

Обсудите функции только для немодулированных программ, подобных загрузчикам.

- * Будьте точными в различиях похожих программ, исходя из условий их использования.

- * Опишите, например, загрузчик связей, его функции.

Перечислите трансляторы в соответствии с обрабатываемыми языками программирования. Дайте уместную информацию об уровне, дополнительных возможностях и ограничениях, которые используются. Если имеется ряд библиотек, посвятите каждой из них подраздел. Опишите их содержание, структуру и все применяемые ограничения. Продолжайте описание всех функциональных и обрабатывающих блоков, подобных, например, служебным программам. Дайте существенную информацию о структуре, возможностях и ограничениях.

- * Сообщите детали о средствах обнаружения ошибок. Обсудите функции, структуру, возможности, ограничения.

Раздел Е: Взаимодействие программиста с системой

Этот раздел будет основным для читателей, которые будут использовать ваше «Руководство» в качестве справочника. Эффективность всего «Руководства» определяется именно этим разделом.

I. Введение к системным связям. Опишите все реальные способы связи между программистом и системой. Укажите пользователю функции управляющих записей, сообщение ввода и работы системы. Сопоставьте средства связи с ситуациями, требующими таких связей. Специфические форматы карт и т. п. дайте в деталях, описывающих связи.

Условные обозначения, используемые для описания таких форматов, являются, однако, частью «Введения».

* Раздел, озаглавленный «Формат, используемый для представления каждой управляющей записи» в этом «Руководстве», был бы здесь весьма полезен. Под таким заголовком опишите способ представления каждой управляющей записи.

Используйте графическое представление, иллюстрируя поля записи. Объясните цель каждого поля записи. Проведите ясное различие между обязательными и необязательными полями.

Представьте условные обозначения. Опишите форму, используемую для представления примеров.

* Перечислите символы, такие, как скобки, и объясните их цель и значение при использовании в тексте.

II. Язык управления заданиями. Этот раздел предназначен для того, чтобы кратко описать те элементы языка управления заданиями, которые жизненно важны для управления данной системой. Только эта информация уместна здесь и должна быть включена в этот раздел. Информация, важная для работы с системой, должна быть ясной и полной. Начните с описания используемых форматов. Проведите различие между обязательными и необязательными полями. Используйте табличную форму для представления данных. Применяйте много иллюстраций. Объясните условные обозначения, используемые в языке.

Перечислите применяемые здесь ограничения. Помогите пользователю узнавать ситуации, в которых действуют ограничения. Расскажите ему, как обойти эти ограничения, если это может быть выполнено путем изменения структуры программы или другими стандартными методами. Следующие два раздела иллюстрируют стандартные ситуации, в которых программист должен установить связь между системой и своей программой, не зависящую от характеристик программы, т. е. они описывают основные ситуации связи, общие для всех программ, использующих систему.

Информация поэтому должна быть представлена в универсальных выражениях. Отличия, необязательные поля и т. д., которые применяются в отдельных случаях, описываются в специальном пункте раздела F при описании связей с различными компонентами.

Взаимодействие и взаимозависимости должны быть описаны здесь с более полным объяснением, даваемым в специальном пункте раздела F.

* Например, если по ходу объяснения встречается ситуация взаимодействия между задачей и монитором, отсылайте к описанию взаимодействия между этими компонентами. Объясните типичный случай, отмечая, что исключения объясняются при их возникновении.

III. Использование мониторов. Этот пункт и все последующие пункты раздела E предназначены для того, чтобы показать использование системы, и должны быть построены таким образом, чтобы пояснять действие системы в работе.

Чтобы быть эффективными в качестве справочного материала, эти пункты должны иметь структуру, указанную ниже.

Начните с вводной части, описывая основные средства связи с рассматриваемыми компонентами системы. Озаглавьте этот раздел так:

а. Как задавать инструкции монитору. В табличной форме перечислите действия управляющих карт и их назначение. Эта часть становится благодаря пояснительному тексту и таблице введением к материалу, который дается ниже.

Теперь начинайте описывать каждую из ситуаций, в которых программист захочет узнать о взаимодействии с системой.

- * Используйте заголовки, отражающие рабочие ситуации.

Затем приведите проиллюстрированное рисунками и примерами описание управляющих записей, необходимых для того, чтобы выполнить задачу обработки.

- * Включите сюда как стандартное употребление управляющих записей, так и случаи их необязательного использования.

- * Дайте список сообщений, содержащий код, номер или текст сообщения, а также смысл и реакцию на такие сообщения.

- * Перечислите ограничения и их действие.

- * Приведите рисунки всех стандартных пакетов перфокарт, показывающие расположение карт в пакете и т. д.

Описанная выше информация могла бы быть представлена под несколькими подзаголовками, зависящими от характеристик рассматриваемой системы.

Вот некоторые из этих заголовков:

b. Как обработать пакет.

c. Как запросить мультиобработку.

d. Как произвести автоматическое планирование и т. д.

Каждое отличное от других содержание работы должно быть предусмотрено и точно описано.

- * Цель этого раздела состоит в том, чтобы дать возможность пользователю узнать свой случай в списке заголовков и перейти непосредственно к инструкции для эффективного использования системы.

Отметим, что инструкция должна указывать последовательность действий для реализации всех возможных случаев, а не учить, когда и почему следует использовать те или иные методы.

IV. Использование загрузчиков. В зависимости от структуры и функций системного загрузчика, наличия более чем одной программы загрузки и т. д. этот раздел может быть достаточно кратким или может состоять из нескольких подразделов.

В относительно простой системе вводная часть:

а. Как задавать инструкции загрузчику — могла бы быть фактически веем, что необходимо. Если описание составлено в соответствии с порядком функционирования, то могут появиться следующие заголовки:

б. Как начать загрузку программы.

с. Как инициировать поиск файла.

д. Как вызывать подпрограммы.

Раздел F: Связь с компонентами специального назначения

Этот раздел дает рекомендации пользователю, основной целью которого является использование возможностей операционной системы для решения некоторой прикладной задачи, такой, как обработка программы, написанной на проблемно-ориентированном языке, генерация проверочных данных, создание библиотеки. Это не значит, что может быть сделана попытка дать пользователю информацию о таких вещах, как характеристики проблемно-ориентированных языков или методы, используемые в библиотеке программ для математиков. Этот материал, по существу, составляет часть документации по определенной компоненте операционной системы.

Этот раздел должен описывать способы, которыми пользователь устанавливает связь с компонентами системы. Таким образом, каждая из компонент специального назначения, т. е. независимые части программного обеспечения, которые входят в рассматриваемую систему, описывается только с точки зрения связей пользователя с операционной системой. Каждый пункт раздела F содержит в себе описание конкретной рабочей ситуации. Для конкретной системы некоторые пункты можно значительно расширить, а некоторые совсем исключить.

1. Использование трансляторов. Начните с пояснения той информации, которая необходима как для использования какого-либо одного, так и для использования всех трансляторов.

Ориентируйте структуру и содержание пункта на особенности рабочих ситуаций.

* Например, используйте такие подзаголовки:

а. Как собрать пакет.

б. Как скомпилировать/выполнить программу.

Некоторые из управляющих записей будут являться вариантами управляющих записей, поясняемых в других разделах.

* Например, монитор может обрабатывать все карты задания, вызывая требуемые компоненты (части ОС) в соответствии с параметрами запроса.

В таких случаях покажите формат управляющей записи. Внесите информацию, необходимую для трансляторов. Ограничьте свое рассмотрение параметрами, используемыми для обращения к трансляторам.

Расскажите пользователю все, что ему необходимо знать для эффективного использования данной системы при выполнении своей программы.

* Например, используйте такие подзаголовки:

Стандартные управляющие записи.

Сообщения.

Ограничения.

Используйте табличную и списочную формы всякий раз, когда сообщаемый материал сам имеет такую структуру. То, что следует за подзаголовками, приведенными выше, иллюстрирует, как выглядит ввод.

Используйте такие подзаголовки:

с. Состав колоды перфокарт. Проиллюстрируйте состав пакета. Дайте рисунки всех различных наборов. Посвятите раздел обсуждению каждого транслятора. Начните с подраздела:

д. Специальные замечания к (название транслятора). После представления общей информации для создания основы опишите замечания, которые должны быть известны пользователю, чтобы использовать некоторый конкретный транслятор.

II. Использование системы отладки. Помните, что этот раздел не предназначен ни для обоснования, ни для инструктирования в использовании методов проверки. Кратко и компактно, как это позволяет материал, приведите стандартные процедуры для пояснения возможностей проверки запросов обработки и их выполнения.

* Используйте такие подзаголовки:

а. Как использовать отладку на исходном языке.

б. Как произвести прикидку времени выполнения.

Под этими подзаголовками опишите:

Управляющие записи.

Состав колоды перфокарт.

Ограничения.

Будьте особенно точными при описании ограничений, которые используются. Приводите рисунки и таблицы везде, где эти формы улучшают представление.

III. Использование библиотек. Опишите использование библиотеки как функцию использования системы.

Организируйте ваш материал так, чтобы он соответствовал специфической ситуации, в которой он используется.

* Например, введите такие подзаголовки:

а. Как вызвать библиотеку (имя библиотеки)

б. Как изменить библиотеку (имя библиотеки)

с. Как подключить дополнительные библиотеки в систему.

* Повторите такие подзаголовки:

Управляющие записи.

Состав колоды перфокарт.

Ограничения.

Используйте таблицы, диаграммы, иллюстрации в той степени, в которой они способствуют представлению материала.

IV. Использование служебных программ. Подготовьте главу для каждого отдельного класса служебных программ.

Давайте пояснения не отвлеченно, а на примерах конкретных рабочих ситуаций.

* Например, используйте такие подзаголовки:

Как использовать данный тип программ (название)

при использовании системы (название).

Включите сюда такую информацию, как:

Управляющие записи.

Состав колоды перфокарт.

Ограничения.

Приложение А: Резюме

Сделайте этот раздел компактным, всесторонним описанием характеристик системы. Широко используйте таблицы и диаграммы. Сгруппируйте параграфы, «Резюме» по подзаголовкам, которые определяют действующие компоненты. Дайте отдельное заключение по каждому различному виду информации, которую пользователь может искать.

- * Например, подведите итог:
 - О возможностях системы.
 - Об управляющих записях.
 - О составе оборудования.
 - О составе программного обеспечения.
 - Об ограничениях.

Приложение В: Указателя

Отметим, что этот раздел отличается от обычно используемых указателей следующим. Указатели, представленные здесь, ограничены характеристиками системы. Они организованы в соответствии с классами информации о системе, а не по алфавиту.

- * Включите в ваши указатели такой часто используемый материал, как:
 - Возможности системы.
 - Управляющие записи.
 - Сообщения системы.
 - Компоненты системы.
 - Рисунки составов колод перфокарт.

Модель 12

ПОДГОТОВКА РУКОВОДСТВА ОПЕРАТОРУ ПО ОПЕРАЦИОННОЙ СИСТЕМЕ

Часть I

- Предисловие (общий раздел) ().
- Эффективное использование этого руководства ().
- Оглавление (общий раздел) ().
- Раздел А: Общая информация ().
 - I. Введение ().

II. Основные понятия операционной системы (название) ().

а. Функции и техническое оснащение системы (Общее представление о работе системы).

1. Аппаратная и программные средства.

2. Пакетная и дистанционная обработка.

III. Структура (название системы) ().

а. Функции.

б. Системные файлы.

IV. Стандарты и условные обозначения ().

Раздел В: (название системы) и аппаратные и программные средства ().

I. Конфигурации оборудования ().

а. Требования к машине ().

1. Центральный процессор.

2. Печать/перфорация.

3. Ленты/диски.

4. Удаленные пульты.

5. Устройства дистанционной связи.

б. Используемое дополнительное оборудование ().

(Повторите перечисленные выше заголовки).

II. Программное обеспечение ().

Мониторы.

Трансляторы.

Служебные программы управления данными.

Служебные программы для наращивания программ.

Раздел С: Возможности, представляемые системой (название системы) ().

I. Режим работы ().

а. Мультипрограммирование.

Действия оператора.

Специальные замечания.

б. Мультиобработка.

Действие оператора.

Специальные замечания.

с. Дистанционная обработка.

Действия оператора.

Специальные замечания.

II. Управление вводом-выводом ().

а. Управление файлами.

б. Средства связи между накопителями.

- III. Средства, облегчающие обработку ().
 - a. Трансляторы.
 - b. Библиотеки.
 - c. Служебные программы.

- IV. Средства обнаружения ошибок ().
 - a. Отладочные пакеты.
 - b. Генераторы проверочных данных.

- V. Вспомогательные средства ().
 - a. Учет.
 - b. Планирование.

Раздел D: Компоненты системы (название системы) ().

- I. Мониторы.
- II. Загрузчики.
- III. Трансляторы.
- IV. Библиотеки.
- V. Средства обнаружения ошибок.
- VI. Служебные программы.
- VII. Вспомогательные средства.
 - a. Управление данными.
 - b. Нарращивание программ.
 - c. Преобразование данных.

Раздел E: Связи программиста с системой ().

- I. Управляющие записи ().
 - a. Имя.
 - b. Функция.
 - c. Формат.
 - d. Расположение.
- II. Сообщения ().
 - a. Номер/код, текст.
 - b. Порождающая компонента.
 - c. Смысл.
 - d. Реакция программиста.

Раздел F: Связи оператора с системой ().

- I. Управляющие записи.
 - a. Имя.
 - b. Функция.
 - c. Формат.
 - d. Размещение.
- II. Сообщения.
 - a. Сообщения о нормальной работе.
 - b. Сообщения об ошибках.

Раздел G: Стандартные действия ().

- I. Разработка процедур.

- а. Управляющие записи.
 - 1. Формат.
 - 2. Стандартные части.
 - 3. Необязательные части.
 - 4. Действие.
 - 5. Размещение.
- б. Сообщения.
 - 1. Номер/код или текст.
 - 2. Смысл.
 - 3. Реакция оператора.
- II. Загрузка системы.
 - а. Начальные процедуры.
 - б. Пультовая подготовка.
 - с. Как загрузить систему.
 - д. Что делать при неудачной загрузке.
- III. Генерация системы.
 - а. Когда система может быть сгенерирована.
 - б. Начальные процедуры.
 - с. Пультовая подготовка.
 - д. Как генерировать.
 - е. Что делать при неудачной генерации.
- IV. Работа системы в режиме пакетной обработки.
 - а. Начало.
 - б. Пультовая подготовка.
 - с. Как управлять пакетным режимом.
- V. Работа системы в мультипрограммном режиме.
 - а. Начало.
 - б. Пультовая подготовка.
 - с. Как управлять заданиями при мультипрограммировании.
- VI. Работа системы в реальном времени.
 - а. Начало.
 - б. Пультовая подготовка.
 - с. Как управлять заданиями в реальном времени.

(Перечисленные выше рабочие ситуации даны как типичное содержание работ. Они могут быть другими для данной системы. Если так, то эти могут не использоваться.)

Раздел Н: Работа с компонентами системы ().

- I. Трансляторы.
 - а. Ассемблер.
 - 1. Начало.
 - 2. Пультовая подготовка.

3. Как управлять сборками.

(Этот раздел повторяется для всех трансляторов в системе.)

II. Служебные программы.

а. Служебные программы управления данными.

1. Начало.

2. Пультовая подготовка.

3. Как управлять программами управления данными.

(Этот раздел повторяется для всех пакетов служебных программ.)

Здесь могут быть написаны дополнительные подразделы, чтобы ссылаться на специальные компоненты операционной системы. Отметим, что применения должны присутствовать в одном из описаний, приведенных в разделе G. Ряд необходимых разделов и окончательный объем информации, которые должны быть представлены, могут указывать на необходимость подготовки отличного от других руководства для некоторых компонент. Индивидуальные руководства должны следовать основной структуре.)

Приложения: Таблицы ().

а. Таблица форматов управляющих карт оператора.

б. Таблица компонент системы.

с. Таблица кодов карт для ввода.

д. Таблица пультовых переключателей и их функций.

е. Таблица двоичных, десятичных, графических и восьмеричных, шестнадцатеричных и т. д. эквивалентов.

Словарь терминов (общий раздел) ().

Часть II

Рекомендации по эффективному использованию руководства

В описании работы с «Руководством» должна быть представлена информация, например, такого характера:

* Обслуживающий персонал включает в себя: персонал по подготовке данных, руководящий и планирующий персонал, персонал по управлению данными и операторов на машинах.

- * Обработка информации для такого персонала включает в себя все действия, которые должны быть предприняты с момента передачи данных пользователем до момента, когда пользователь получает ответ.

Адресуйте ваше руководство каждой группе персонала с их специфическими интересами. Придавайте особое значение разделам, посвященным тому, как выполнить ту или иную операцию.

Отсылайте читателя к «Оглавлению» для поиска раздела, который дает необходимую ему информацию.

Укажите точно, какие требуются основные знания об оборудовании.

Раздел А: Общая информация

I. Введение. Введите основные понятия, которые являются фундаментальными для понимания остальной части «Руководства». Обращайтесь во «Введении» ко всем читателям в соответствии с их подготовкой. Свободно используйте подзаголовки, чтобы помочь читателю в нахождении им своего уровня. Организуйте «Введение» в соответствии с основными подзаголовками и разверните по мере необходимости каждую тему.

Адресуйте ваше обсуждение к оператору; укажите на его роль в обработке данных и на необходимость точного соблюдения инструкций.

II. Основные понятия операционной системы (название системы).

а. *Функции и техническое оснащение системы.* Опишите специфические функции, влияющие на оснащение данной системы.

- * Свяжите этот раздел с описанием общей идеологии операционной системы.

Организируйте этот материал так, чтобы он содержал характеристики описываемой системы.

- * Перечислите, например, возможные конфигурации оборудования.

- * Если речь идет об обработке, четко отличайте дистанционную обработку от пакетной.

III. Структура (название системы). Проиллюстрируйте внутреннюю структуру системы, перечислите модули, которые составляют систему. Покажите, как связаны

эти модули, как они взаимодействуют, чтобы выполнить различные функции системы. Приведите блок-схемы, показывающие различные комбинации, если такое графическое представление улучшает понимание. Перечислите файлы системы и дайте их характеристики, такие, как носители, размещение, устройства, размеры и т. д.

IV. Стандарты и условные обозначения. Определите все стандартные термины, которые используются в «Руководстве».

Отметьте, что это определенное значение, вероятно, имело основу в традиционном использовании или отражает некоторую специальную характеристику системы.

В вашем тексте стандартное использование технических терминов должно стать нормой.

Объясните все условные обозначения, которые используются в описании системы.

Используйте этот раздел также и для того, чтобы описать формат представления: Объясните графически, как общие вопросы, касающиеся управляющих карт, сообщений и т. д., будут представлены в последующих разделах.

Здесь был бы важным раздел, озаглавленный «Использование форматов для представления каждой управляющей записи в этом руководстве». Под этим заголовком опишите способ представления каждой управляющей записи.

Используйте графическое представление, иллюстрируя поля управляющих записей.

Объясните назначение каждого поля записи.

Делайте четкое различие между необходимыми и необязательными полями.

Дайте список условных обозначений. Опишите форму, используемую для приведения примеров.

Перечислите символы (такие, как скобки) и объясните их цель и значение при использовании в тексте.

Раздел В: (название системы) и аппаратные и программные средства

Опишите систему в работе.

I. Конфигурации оборудования. Начните с описания основных составных частей машины, необходимых для использования системы.

а. Требования к машине. Перейдите от наиболее элементарных требований к оборудованию и связанных с ним функций системы к наиболее сложным конфигурациям. Перечислите те части, для которых возможны отклонения: центральный процессор, устройства чтения карт, печать/перфорация, ленты/диски, выносные пульта и устройства дистанционной связи.

б. Используемое дополнительное оборудование. Делайте различие между стандартными и расширенными конфигурациями.

Примером последних могло бы быть наличие удаленных пультов как специальных устройств для обеспечения расширенного использования.

- * Посвятите подраздел каждой возможной конфигурации.
- * Укажите границы выполняемых изменений, допускаемых расширеними.
- * Опишите отличия в работе различных возможных конфигураций.

Рекомендуется использовать табличную форму представления этого материала везде, где это возможно.

II. Программное обеспечение. Представьте как единое целое все программное обеспечение.

- * Опишите рабочие функции различных категорий программного обеспечения. Будьте кратки и избегайте нетехнических терминов.

Когда основной стержень разработан, опишите части программного обеспечения в каждой категории. Используйте подзаголовки для указания категорий.

- * Например: мониторы, трансляторы, служебные программы управления данными, служебные программы для наращивания программ и т. д.

Проиллюстрируйте взаимодействие компонент, используя блок-схемы.

Раздел С: Возможности, предоставляемые системой (название системы)

Приведите все возможности системы с точки зрения их влияния на последовательность и процедуру обработки. Организуйте и сгруппируйте информацию

В этом разделе в соответствии с рабочими ситуациями.

* Так, например, предлагаемые подразделы могли бы иметь следующий вид.

I. Режимы работы. Опишите возможные режимы работы, такие, как мультипрограммирование, мультиобработка, дистанционная обработка и т. д. Рассмотрите эти режимы с точки зрения выбора оборудования. Перечислите допустимые изменения в оборудовании, необходимые для различных рабочих ситуаций.

Используйте такие подзаголовки, как: «Специальные замечания», «Действия оператора» и т. п., для того, чтобы идентифицировать информацию.

II. Управление вводом-выводом. Примерами информации, которая должна быть приведена здесь, являются системное обеспечение для управления файлами и средства связи для обмена между накопителями.

III. Средства, облегчающие обработку. Разделите этот материал по функциональному назначению, чтобы дать оператору понятие о целях таких программ, как трансляторы, библиотеки, служебные программы и т. д.

* Объясните кратко функции каждой категории.

Сделайте ударение на сходстве работы с элементами данной категории.

IV. Средства обнаружения ошибок. Опишите кратко средства обнаружения ошибок, предоставляемые системой.

Этот раздел, так же как и предшествующий, должен быть кратким. Цель его состоит в том, чтобы дать возможность оператору ориентироваться в системе и в то же время дать ему некоторое представление о рациональности различных последовательностей его действий, действий системы и т. д., которые относятся к различным рабочим ситуациям.

* Перечислите программы в соответствии с их функциональными категориями, такими, как;

а. Отладочные пакеты.

б. Генераторы проверочных данных.

V. Вспомогательные средства. Этот раздел будет иметь большее значение для оператора, чем все

предыдущие, так как он будет привлекаться к использованию таких возможностей системы, как учет и планирование в системе.

Объясните, что делается системой для упрощения таких работ.

Раздел D: Компоненты системы (название системы)

В этом разделе опишите компоненты системы, которые обеспечивают возможности, описанные в разделе C.

Дайте названия программ, при помощи которых делаются обращения к компонентам в операционной системе.

Разбейте эту информацию по функциональным категориям. Заголовками могут быть следующие:

- I. Мониторы.
- II. Загрузчики.
- III. Трансляторы.
- IV. Библиотеки.
- V. Средства обнаружения ошибок.
- VI. Служебные программы.
- VII. Вспомогательные средства.

Используйте подзаголовки, чтобы отличать программы внутри каждой группы.

* Например, служебные программы можно было бы дальше подразделить следующим образом:

- a. Управление данными.
- b. Нарастивание программ.
- c. Преобразование данных.

Везде, где возможно, рекомендуется табличная форма, показывающая название компонент, системные метки и функции.

Раздел E: Связи программиста с системой

Материал, представленный в этом разделе, предназначен только для справок; здесь не должно быть ни детального, ни длительного объяснения.

I. Управляющие записи. Цель информации, даваемой здесь оператору, состоит в том, чтобы дать возможность проверить пульта и т. д., предназначенные для работы. Здесь должны использоваться только ос-

новые понятия, такие, как формат, расположение и назначение управляющих записей.

Должны использоваться таблицы и диаграммы, чтобы представить информацию графически и кратко, как только возможно.

* Например, могла бы быть использована таблица следующего вида:

Управляющие карты

Название	Функция	Формат

Расположение записей во входном потоке лучше иллюстрировать серией рисунков.

II. Сообщения. Этот раздел должен начинаться с ясного указания, кому направлены эти сообщения, в данном случае — программисту.

Они приводятся здесь для того, чтобы дать оператору полную картину стандартного содержания работы. Табличная форма рекомендуется везде, где возможно.

* Например, информация в этом разделе могла бы быть сведена в следующую таблицу:

Номер/код/текст	Чье сообщение	Смысл сообщения	Реакция программиста

Раздел F: Связи оператора с системой

Этот раздел и разделы «Стандартные действия», «Работа с компонентами системы» и «Таблицы» создают основу справочного материала, которым оператор будет чаще всего пользоваться.

Информацией, приводимой в этих разделах, оператор будет пользоваться ежедневно. Поэтому она должна быть представлена выразительно и наглядно.

Широко используйте таблицы, диаграммы, рисунки, чертежи и т. д.

Информация о связях оператора с системой аналогична такой же информации, предназначенной для программиста. Она дает оператору основные средства для связи с системой.

Все управляющие карты оператора должны быть пояснены. Форматы, показывающие стандартное и необязательное использование, должны быть начерчены.

Диаграммы, иллюстрирующие расположение управляющих карт, должны быть приведены для каждого варианта пакета.

Следует соблюдать осторожность при начертании расположений управляющих карт и пакетов. Попытка изобразить несколько ситуаций на одном рисунке или на одной и той же странице может привести к путанице. Предпочтительно вводить больше страниц диаграмм, ясно помеченных и расположенных в порядке возрастания сложности.

Говорите прямо и ясно. Пишите эти разделы так, чтобы изучающий их персонал мог использовать их, если пульта управления станут их рабочим местом. Здесь должны также поясняться основные форматы, используемые в сообщениях, и ответная реакция на них оператора.

Следует делать различие между сообщениями о нормальной работе и сообщениями об ошибках, которые могут потребовать специальных действий.

Иллюстрируйте графически формат, используемый для представления сообщений.

Используйте графическую форму подобно тому, как она использовалась в разделе Е. Отсылайте к разделу «Реакция оператора», а не к «Реакция программиста».

Следует помнить, что сноски, чертежи и рисунки должны быть снабжены номерами и подписями для включения в «Оглавление» и «Резюме».

Раздел G: Стандартные действия

Организируйте содержание этого раздела в соответствии с естественной последовательностью действий оператора.

Начните со стандартных действий, таких, как регистрация и загрузка системы, затем переходите к общим ситуациям, таким, как генерация системы, работа системы в пакетном режиме, с мультипрограммированием, в реальном времени.

Используйте подход поваренной книги, т. е. дайте точный перечень необходимых материалов и набор директив, выполняемых одна за другой для получения требуемых результатов.

Опишите начальную и пультовую подготовку.

Опишите управляющие записи, стандартные части, необязательные части, их действие и размещение.

Обсудите сообщения. Делайте ссылки на номер, код или текст, смысл, реакцию.

Если необходимо, включите сюда схемы оборудования с ясно помеченными индикаторами, условиями, переключателями и т. д. Используйте блок-схемы, если такая форма будет способствовать пониманию.

Рассмотрите использование таблицы решений, чтобы показать решение многими способами. Выражайтесь просто, ясно, полно, точно. Оператору надо дать недвусмысленные правила для действий в любой ситуации. Используйте подзаголовок: «Как (сделать что-либо)», чтобы сгруппировать перечни действий.

* Например, озаглавьте параграф, посвященный загрузке системы: Как загрузить систему?

Точно указывайте действия в ситуациях, которые могут возникнуть. Используйте заголовки: «Что делать...».

Например, может быть использован параграф, озаглавленный: «Что делать при неудачной генерации».

Описывайте различные рабочие ситуации в том виде, в каком с ними сталкивается оператор.

Например, используйте такие заголовки.

Работа системы в пакетном режиме

Работа системы с мультипрограммированием

Тон здесь особенно важен; оператора следует почти буквально вести за руку, но он не должен чувствовать, что его хотят перекричать.

Раздел Н: Работа с компонентами системы

Здесь приводится информация для использования каждого класса программ специального назначения. Эта информация подобна той, которая дается для стандартных рабочих ситуаций. Каждой категории следует посвятить раздел, такой, как трансляторы, служебные программы и т. д. — группы, описанные в разделе D. Дайте полную информацию о подготовке и начале работы. Опишите пультовую подготовку.

Здесь настоятельно рекомендуются подзаголовки: «Как сделать...», чтобы дать руководство к работе.

Оба эти раздела G и H содержат переменное число подразделов. Объем информации, которая должна приводиться, определяет сложность рассматриваемой системы.

Приложение: Таблицы

Здесь должны быть повторены таблицы, которые появляются в тексте, которыми часто пользуются и на которые ссылаются.

* Например, таблица форматов управляющих карт оператора должна быть включена в этот раздел наряду с таблицей компонент системы.

Должны быть подготовлены для включения в этот раздел дополнительные таблицы с информацией, представляющие интерес для оператора.

* Например, таблицы двоичных и восьмеричных эквивалентов, представление букв и шестнадцатеричных эквивалентов, коды карт для ввода и переключатели пульта управления и их функции.

Характеристики рассматриваемой системы являются определяющим фактором.

Модель 13

ПОДГОТОВКА ОПИСАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Часть I

Предисловие (общий раздел) ().

Рекомендации по эффективному использованию этого описания ().

Оглавление (общий раздел) ().

Раздел А: Общая информация ().

I. Введение ().

Раздел В: Операционные системы ().

I. Введение.

II. (Название первой операционной системы) } Повторите для каждой операционной системы.

Раздел С: Трансляторы ().

I. Введение.

II. (Название первого транслятора.) } Повторите для каждого транслятора.

Раздел D: Программы обеспечения совместимости ().

I. Введение.

II. (Название первой программы.) } Повторите для каждой программы.

Раздел Е: Библиотеки ().

I. Введение.

II. (Имя первой библиотеки) } Повторите для каждой библиотеки.

Раздел F: Служебные программы ().

I. Введение.

II. (Название первой служебной программы.) } Повторите для каждой группы служебных программ.

Раздел G: Прикладные программы ().

I. Введение.

II. (Название первого типа прикладных программ.) } Повторите для каждого пакета прикладных программ.

Раздел H: Пакеты программ общего пользования ().

I. Введение.

II. (Название первой программы.) } Повторите для каждого пакета.

Приложения: Взаимодействие программ ().

а. Структура ().

б. Зависимости ().

Словарь терминов (общий раздел) ().

Часть II

Рекомендации по эффективному использованию этого описания

Инструкция по составу программного обеспечения должна снабдить читателя указаниями по размещению информации:

- * о возможностях системы и вспомогательном оборудовании, предоставляемых изготовителем;
- * о помощи в планировании использования оборудования;

* об оценке усилий, требуемых для выполнения различных работ.

Обратите внимание читателя на все возможности предоставляемого оборудования.

Раздел А: Общая информация

1. Введение. В этом разделе представьте полный набор систем программирования для ряда машин этого семейства.

Сравните различные системы программирования.

* Если возможно, дайте графическую иллюстрацию этого.

З а м е ч а н и е. Разделы с А до F описывают системы программного обеспечения, т. е. такие программы и системы, как операционные системы, трансляторы и т. д., которые помогают программисту в подготовке рабочей программы.

Используйте указанные заголовки разделов.

* Отметим, что эти заголовки располагаются в порядке важности, а не в алфавитном порядке.

* Свободно придумывайте некоторые дополнительные заголовки для программ или систем, помимо тех, которые уже описаны; вносите эти заголовки на соответствующие места.

Раздел В: Операционные системы

Операционная система управляет потоком заданий и распределяет ресурсы оборудования и программного обеспечения.

1. Введение. Кратко опишите операционную систему или систему, имеющуюся на этой машине или ряде машин.

II. (Название первой операционной системы). Опишите особенности и возможности этой системы.

Сравните ее с другими операционными системами для того же оборудования.

* Отсылайте читателя к руководству или руководствам, которые описывают данную систему в деталях.

Укажите требования к оборудованию со стороны операционной системы:

- * Если возможна более чем одна версия данной системы, то опишите требования к оборудованию для каждой версии.
- * Опишите пониженные возможности ограниченной версии операционной системы, которая требует меньше оборудования.

Раздел С: Трансляторы

I. Введение. Перечислите компиляторы, макрогенераторы, ассемблеры и т. д., которые имеются на машине или ряде машин.

II. (Название первого транслятора). Опишите особенности и возможности данного транслятора.

Сравните его язык с другими версиями того же языка, с подобными языками.

Перечислите специфические особенности, которые отличают его (транслятор) от других.

- * Отсылайте читателя к руководствам, которые описывают языки программирования в деталях.

Опишите взаимодействие транслятора с имеющимися операционными системами.

Укажите требования к оборудованию со стороны транслятора, включая все имеющиеся версии этого транслятора.

Опишите пониженные возможности ограниченной версии транслятора, которая требует меньше оборудования.

Отсылайте читателя к имеющимся руководствам для пополнения информации.

Раздел D: Программы обеспечения совместимости

I. Введение. Перечислите все такие программы, включая интерпретаторы, моделирующие программы, загрузчики и т. д., которые позволяют программы, написанные для других машин, считать на этой машине.

II. (Название первой программы). Опишите зависимость этой программы от других частей программного обес-

печения. Дайте критерий эффективности этой программы совместимости.

Опишите взаимодействие программы с имеющимися операционными системами. Укажите требования к оборудованию со стороны программы.

Отсылайте читателя к имеющейся литературе по этому вопросу.

Раздел Е: Библиотеки

I. Введение. Укажите, какие имеются библиотеки программ, и опишите их очень кратко. Представьте библиотеки под подзаголовками, такими, как: «Математическая библиотека», «Статистическая библиотека» и т. д.

II. (Имя первой библиотеки). Кратко опишите использование этой библиотеки и ее общий вид.

Перечислите имеющиеся в библиотеке программы с описанием выполняемых функций. Отсылайте читателя к имеющейся литературе по этим вопросам.

Раздел F: Служебные программы

I. Введение. Кратко опишите все имеющиеся служебные программы, такие, как программы печати, программы переписи с карт на ленту, программы вывода и т. д.

II. (Первая служебная программа). Опишите особенности и возможности программы. Отсылайте читателя к имеющимся руководствам для пополнения информации.

З а м е ч а н и е. Разделы G и H описывают проблемно-ориентированные блоки программного обеспечения.

Раздел G: Прикладные программы

I. Введение. Приведите общее описание типов имеющихся прикладных программ, таких, как программы ведения описей, программы ведения платежных ведомостей, программы определения критического пути и т. д.

Используйте типы программ для того, чтобы сгруппировать имеющиеся программы по следующим разделам.

II. (Название первого типа прикладных программ). Кратко опишите особенности и возможности этих программ.

* Отметьте все частные особенности.

* Отметьте все ограничения.

Отсылайте читателя к имеющейся литературе по этим вопросам.

Раздел Н: Пакеты программ общего пользования

I. Введение. Кратко опишите имеющиеся пакеты программ общего пользования, такие, как генераторы отчетов, программы сортировки, программы слияния, программы вывода, графиков и т. д. Используйте типы программ в качестве подзаголовков для группировки программ в следующих разделах.

II. (Название первого типа программ общего пользования). Укажите кратко возможности каждой из этих программ.

* Отметьте все специфические особенности.

Отсылайте читателя к имеющейся литературе по этим вопросам.

Приложения: Взаимодействие программ

а. Структура. О ряде программ программного обеспечения можно говорить как о едином целом, если имеются взаимосвязи, например, если программа X не может быть использована без программы Y. Используйте диаграммы, чтобы показать взаимосвязи, существующие в программном обеспечении.

б. Зависимости. Начертите таблицу, в которой в первом столбике перечислены основные программы, во втором — программы, которые им требуются, в третьем — которым они требуются.

* Например, можно привести следующую таблицу вхождений:

Программа	Вызываемые программы	Вызывающие программы
Антилогарифм	Логарифм	Степень

Модель 14

ПОДГОТОВКА ПРОСПЕКТОВ

Часть I

Предисловие (общий раздел) ().

Оглавление (общий раздел) ().

I. Описание ().

II. Особенности ().

III. Функции ().

IV. Готовность ().

V. Дополнительная информация ().

Часть II

«Рекомендации по эффективному использованию» не являются необходимыми в «Проспекте». Материал представляется в общем и кратком виде; основное направление использования этого проспекта состоит в ознакомлении с системой, описываемой им.

Предисловие можно использовать здесь для того, чтобы создать некоторую основу для описания системы. Однако краткость и образность являются лейтмотивом.

В коротких предложениях документа должны использоваться примеры и стандартная терминология.

I. Описание. Опишите в нескольких коротких параграфах коммерческие или инженерные применения системы. Дайте широкое описание, так чтобы покупатели или пользователи поняли, как эта или подобная ей система будет удовлетворять их собственные потребности.

Будьте внимательны, чтобы ограничить это описание (приблизительно 150 слов является хорошим правилом). Занятый покупатель или пользователь не будет иметь времени или желания, чтобы просмотреть длинный материал для того, чтобы определить, интересуется ли его система или нет.

* Помните, что покупатель или пользователь не станет читать дальше, если не пробудить здесь его интерес.

Возьмите себе за правило определять любой термин, который вы используете, сразу же при его появлении. Читатель должен в этом типе руководства сразу же уловить важность того, что вы говорите, без обращения к словарю терминов или к специалисту в данной области.

II. Особенности. Перечислите и объясните в одном или двух предложениях основные особенности системы. Особенности являются теми факторами, которые делают систему особенно полезной и ценной для пользователей. Пять — семь характерных черт являются самым большим, что вы должны попытаться перечислить здесь; поэтому выберите их тщательно и посмотрите, насколько важны они будут для покупателей или пользователей.

III. Функции. Перечислите и посвятите несколько предложений описанию каждой из функций, которые система выполняет. Ограничьте этот перечень не более чем десятью наименованиями.

- * Перечислите эти функции в том порядке, в котором их выполняет система, и в описании каждой из них расскажите, как она связана с остальными, чтобы читатель получил общее представление о том, что система способна делать.

Вы не можете в «Описании» рассказать о системе в деталях. Поэтому используйте представившуюся возможность, чтобы рассказать о функциях, которые система выполняет, расширить описание и дать читателю более полную картину системы.

- * Не поддавайтесь, однако, побуждению встать на путь научной трактовки всего, что связано с функциями, которые выполняет система.

Помните, что в документах такого типа вы не сможете никогда сделать больше, чем дать основную, общую информацию.

- * Постарайтесь объяснить каждую функцию таким образом, чтобы читатель смог сопоставить ее с подобной или похожей функцией, выполняемой в настоящее время в его сфере работы неавтоматически, или, по крайней мере, дайте ему понять на словах, как применение системы могло бы способствовать повышению эффективности работы, выполняемой в его сфере деятельности.

Здесь должны приводиться в высшей степени простые рисунки, которые показывают главные входы и выходы, порожденные системой, и принципиальные элементы или части системы. Такие рисунки стали бы удобным средством подытожить и связать перечисленные и рассмотренные функции.

IV. Готовность. Сообщите приблизительные или точные сроки, в которые прикладная система будет создана, проверена и готова к использованию.

V. Дополнительная информация. Укажите лицо, учреждение или отдел, с которыми заинтересованные покупатели и пользователи могут вступить в контакт для того, чтобы получить дальнейшую более детальную информацию, касающуюся применений данной системы или вообще применений систем и программ.

ПРИЛОЖЕНИЕ

КАК ПИСАТЬ РУКОВОДСТВО, ИСПОЛЬЗУЯ МОДЕЛЬ ДОКУМЕНТА

В приводимом ниже тексте задача написания показана в соответствии со стадиями ее выполнения.

Подготовка

- * Соберите все исходные материалы — независимо от того, являются ли они вашими собственными заметками, описаниями и т. д. или они представляются другими авторами.
- * Прочтите эти материалы. Ознакомьтесь с их содержанием и способом изложения. Отметьте все термины или темы, которые требуют пояснения.
- * Соберите весь дополнительный материал, который может быть использован.
- * Используя таблицу (стр. 4), составьте перечень всех групп персонала, который составляет аудиторию читателей.
- * Представьте себя в роли каждого из них. Поймите, что вы хотите от того вида документа, который вы подготавливаете.

Если вы недостаточно знаете обязанности одного из ваших читателей, выясните их прежде, чем будете писать документ. Если необходимо, опросите читателей, к которым вы обращаетесь.

Набросок содержания

- * Подготовьте план вашего руководства, т. е. напишите заголовки.
- * Внимательно просмотрите заголовки; укажите в верхней части каждой страницы группы персонала, которым будет требоваться информация, приводимая под данным заголовком.
- * Перечислите группы персонала, к которым вы собираетесь адресоваться.

- * Перечислите для каждой группы персонала разделы документа, которые будут ее интересовать.
- * Напишите «Рекомендации по эффективному использованию» вашего документа, используя таблицу (стр. 4).

Заполнение наброска

- * Прочтите исходный материал.
- * Напишите вводные разделы вашего документа.
- * Введите исходную информацию в ваш план документа.
- * По мере того, как вы вводите информацию в план, делайте пометки на верху страниц, указывающие, является ли информация полной.
- * Просмотрите материал, который вы не использовали. Оцените его тщательно, чтобы определить, является ли эта информация посторонней.
- * Просмотрите документ таким образом, чтобы определить, можете ли вы полностью закончить разделы.
- * Напишите те разделы, которые вы можете подготовить к настоящему моменту, приспособляя ваш стиль изложения к аудитории читателей.
- * Найдите недостающую информацию.
- * Закончите написание руководства.

Подготовка к выпуску

- * Прочтите документ, который вы написали, ставя себя по очереди на место каждого читателя.
- * Последуйте своим собственным указаниям, т. е. проверьте ваши рекомендации.
- * Перепишите те разделы, которые требуют изменения стиля изложения, ориентации и т. д., для того, чтобы удовлетворить требования читателей.
- * Проверьте весь текст, чтобы определить, могут ли быть использованы рисунки, диаграммы и/или примеры для того, чтобы заменить повествование (текст).
- * Напишите «Резюме» и «Указатели».
- * Проверьте «Словарь терминов» и исправьте его так, чтобы ввести все термины, которые должны быть определены.

46 коп.

