

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
ГЛАВНОЕ УПРАВЛЕНИЕ ПО ОБРАЗОВАНИЮ
МОГИЛЕВСКОГО ОБЛАСТНОГО ИСПОЛНИТЕЛЬНОГО КОМИТЕТА
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«МОГИЛЁВСКИЙ ГОСУДАРСТВЕННЫЙ ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ»

Учебная дисциплина

Основы алгоритмизации и
программирования

УТВЕРЖДАЮ
Заместитель директора
по учебной работе
_____М.М. Федоськова

КУРСОВОЙ ПРОЕКТ
РАЗРАБОТКА ПРОГРАММНОГО МОДУЛЯ
МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ

Разработчики: преподаватели цикловой комиссии спецдисциплин специальности «Программное обеспечение информационных технологий», Кашпарова Ю.А., Карманов А.В., Пикулина А.Н.

Рецензент: Литвинова Л.Е., преподаватель учреждения образования «Могилевский государственный политехнический колледж»

Аннотация: методические рекомендации разработаны на основании учебной программы учреждения образования «Могилевский государственный политехнический колледж», утвержденной директором колледжа 29.08.2016. Методические рекомендации предназначены для учащихся в процессе выполнения курсового проекта для систематизации знаний, полученных при изучении учебной дисциплины, приобретения опыта в комплексной разработке и реализации практических задач, совершенствования практических навыков в использовании и разработке современных информационных систем.

Обсуждено и одобрено
на заседании цикловой комиссии спецдисциплин
специальности «Программное обеспечение
информационных технологий»

Протокол № ____ от _____

Содержание

Введение	4
1 Общие методические рекомендации	5
1.1 Цели и задачи курсового проекта	5
1.2 Содержание и объем курсового проекта	6
1.3 Порядок работы над курсовым проектом и его защиты	6
2 Методические рекомендации по содержанию пояснительной записки	8
2.1 Введение	8
2.2 Общая характеристика задачи	8
2.3 Проектирование программного модуля	8
2.4 Реализация программного модуля	10
2.5 Мероприятия по энерго- и ресурсосбережению	12
2.6 Заключение	12
3 Оформление курсового проекта	13
3.1 Оформление графической части	13
3.2 Оформление пояснительной записки	13
Список используемых источников	15
Приложение А (обязательное) Пример оформления титульного листа	16
Приложение Б (обязательное) Пример оформления содержания	17
Приложение В (обязательное) Пример оформления бланка задания	18
Приложение Г (информационное) Основные символы схемы алгоритма программы	20
Приложение Д (информационное) Пример схемы алгоритма	23
Приложение Е (информационное) Пример оформления фрагмента программного кода	23

Введение

Методические рекомендации разработаны для учащихся специальности 2-40 01 01 «Программное обеспечение информационных технологий» и содержат требования к выполнению курсового проекта по учебной дисциплине «Основы алгоритмизации и программирования».

Учебная дисциплина «Основы алгоритмизации и программирования» ставит своей целью изучение основ алгоритмизации, методов проектирования и разработки программ на языках программирования Паскаль на основе интегрированной среды Турбо-Паскаля и современной системы программирования Delphi.

В процессе разработки курсового проекта необходимо опираться на сведения, полученные на других специальных курсах, таких как «Математика», «Теория вероятности и математическая статистика», «Информатика».

В качестве базового языка для изучения основ программирования выбран Паскаль по следующим причинам:

- относительно небольшое количество базовых конструкций;
 - структурированность языка;
 - четкое разделение секций описания и реализации;
 - широкие возможности для написания самых различных по направленности программ;
 - гибкие возможности в отношении представления данных.
- система программирования Delphi - логическое продолжение языка Object Pascal и поэтому является хорошим средством разработки объектно-ориентированных приложений.

1 Общие методические рекомендации

1.1 Цели и задачи курсового проекта

Эффективное использование компьютеров для решения инженерных и научных задач невозможно без знаний основных методов составления схем алгоритмов, написания эффективного программного обеспечения на языке программирования.

Целью курсового проекта является обобщение, углубление и закрепление знаний, полученных при изучении учебной дисциплины «Основы алгоритмизации и программирования», а также формирование навыков самостоятельно разрабатывать и создавать программные продукты, проводить их тестирование, работать со справочной и технической литературой.

Задачами курсового проектирования являются:

- систематизация знаний, полученных при изучении учебной дисциплины;
- приобретение опыта в комплексной разработке и реализации практических задач;
- совершенствование практических навыков при разработке современных информационных систем;
- практическое использование языков программирования Паскаль и Delphi, использование OLE и ActiveX технологий для связи объектов различных приложений.

Перед выполнением курсового проекта учащийся должен знать:

- основные этапы проектирования, разработки, отладки, тестирования и верификации программ;
- методы и способы разработки надежных программ;
- способы описания и представления алгоритмов;

После выполнения курсового проекта учащийся должен знать:

- эффективные средства языка для написания программ;
- современные подходы к проектированию и программированию;
- современные методы разработки и создания программных продуктов;
- современные методы и способы тестирования программ;

Перед выполнением курсового проекта учащийся должен уметь:

- разрабатывать стандартные алгоритмы;
- использовать все возможные средства для автоматизации программирования;
- находить различие между структурным и объектным подходом к реализации задачи;

После выполнения курсового проекта учащийся должен уметь:

- разрабатывать алгоритм программы;
- описывать алгоритм программы при помощи языка Delphi;
- применять эффективные средства языка для написания программ;
- строить программы исходя из современных подходов к проектированию и программированию;
- применять различные структуры данных;
- самостоятельно разрабатывать и создавать программные продукты;

- проводить тестирование программ;
- самостоятельно работать со справочной и технической литературой;
- оформлять пояснительную записку и графическую часть курсового проекта в соответствии с применяемыми стандартами;
- кратко, четко, грамотно и последовательно уметь излагать свои мысли, докладывать о процессе и результатах работы над курсовым проектом.

В целом курсовой проект должен дать учащимся приобретение опыта самостоятельной разработки прикладного программного обеспечения для решения расчетных задач для персонального компьютера, который в дальнейшем потребуются им в непосредственной работе по специальности.

1.2 Содержание и объем курсового проекта

1.2.1 Объем курсового проекта:

- пояснительная записка объемом 20-25 листов формата А4;
- графическая часть объемом 2 листа формата А1.

1.2.2 Содержание пояснительной записки включает:

- введение;
- общая характеристика задачи;
- проектирование программного модуля;
В разделе указывается:
 - 1) Описание входных и выходных данных;
 - 2) Алгоритм решения задачи.
- реализация программного модуля;
В разделе указывается:
 - 1) Описание структуры разрабатываемой задачи
 - 2) Проектирование и реализация интерфейса задачи
 - 3) Разработка справочной системы
 - 4) Тестирование задачи
- мероприятия по энерго- и ресурсосбережению;
- заключение;
- список используемых источников;
- приложение.

1.2.3 Содержание графической части.

На листе 1 размещаются блок-схемы алгоритма основных процедур и функций разработанного программного модуля. На листе 2 размещаются входные и выходные формы. Возможно также размещение схемы данных или схемы взаимодействия программ.

Графическая часть проекта выполняется в соответствии с требованиями стандартов на листах стандартных форматов.

1.3 Порядок работы над курсовым проектом и его защиты

1.3.1 Курсовой проект выполняется на основании индивидуальных заданий, выданных преподавателем. В задании на курсовое проектирование указывается

тема курсового проекта, исходные данные, календарный график работы над проектом и срок сдачи курсового проекта.

Курсовой проект выполняется путем самостоятельной работы учащихся с использованием методических рекомендаций по выполнению курсового проекта, специальной литературы, консультаций преподавателя-руководителя курсового проекта. В целях осуществления контроля за ходом выполнения курсового проекта учащиеся предоставляют выполненные разделы проекта преподавателю согласно календарного графика работы.

1.3.2 На защите курсового проекта учащиеся делают доклад, в котором должны быть:

- сформулирована тема курсового проекта и кратко изложена его цель и содержание;

- указаны в краткой форме примененные методы расчетов;

- выводы по работе в форме конкретных предложений.

Доклад должен быть рассчитан на 5-7 минут.

Преподаватель оценивает курсовой проект по следующим критериям:

- работоспособность и корректность программы и правильность расчетов;

- качество оформления пояснительной записки и графической части в соответствии с СТУ 01-32-2017.

- правильность и полнота ответов на поставленные вопросы при защите курсового проекта;

- содержательность доклада учащегося, умение кратко, грамотно доложить сущность курсового проекта.

1.3.3 Критерии оценки выполнения курсовых проектов учащихся по учебной дисциплине «Основы алгоритмизации и программирования» приведены в учебной программе, утвержденной директором колледжа 29.08.2016. Отметка за курсовой проект выставляется с учетом качества выполнения проекта и по итогам защиты. Положительной минимальной отметкой является балл 4 (четыре).

К защите должно быть представлено:

- программный модуль;

- графическая часть;

- пояснительная записка.

2 Методические рекомендации по содержанию пояснительной записки

2.1 Введение

Во введении отражаются:

- состояние и перспективы развития современных информационных технологий;
- место, роль, актуальность использования систем программирования в современных информационных технологиях;
- предназначение и возможности программного обеспечения и пакетов прикладных программ и результаты внедрения их в различных областях жизнедеятельности человека;
- цель данного курсового проекта.

2.2 Общая характеристика задачи

В данном разделе указывается общий метод решения задач данного класса. При необходимости в разделе выполняется изображение рисунков для пояснения принципа решения поставленной задачи.

В данном разделе кратко излагается математический метод решения поставленной задачи с записью необходимых математических выражений и действий. При этом следует обязательно указывать литературные источники, используемые при выполнении математического описания задачи. Все записанные математические формулы и выражения должны иметь нумерацию, выполняемую в круглых скобках в правой части страницы, для организации последующих ссылок на них в тексте пояснительной записки. При необходимости можно самостоятельно получить расчетные формулы, в этом случае указывается метод вывода полученных выражений.

В данном разделе производится определение состава и вида исходных данных, необходимых для решения задачи с помощью персонального компьютера. К ним относятся выражения исходных уравнений или функций, начальные условия или границы интервала, точность или величина шага итераций.

В данном разделе приводятся общие требования к разрабатываемому программному обеспечению, то есть в каких операционных средах должна работать программа, а также какие операционные системы и аппаратные средства имеют пользователи.

2.3 Проектирование программного модуля

2.3.1 Описание входных и выходных данных. Прежде чем приступить к построению схемы алгоритма, предварительно следует определить следующие данные:

- исходные (входные) данные, которые должны вводиться с клавиатуры или из файла;

- промежуточные результаты, необходимые для реализации вычислительного процесса, их результаты расчета непосредственно в программе не должны отображаться, но могут использоваться при ее отладке и верификации;

- выходные результаты решения, которые должны отображаться на экране или выводиться в текстовый файл.

При задании состава переменных следует всегда сохранять вводимые значения исходных данных до конца программы, не допуская их переопределения в результате расчетов.

Если, согласно задания на курсовой проект, используется объектно-ориентированная технология программирования, тогда в этом разделе следует представить собственную иерархию классов. При этом число уровней наследования классов должно быть не менее двух. В иерархии классов должны быть максимально представлены основные свойства объектно-ориентированного программирования: инкапсуляция, наследование и полиморфизм классов.

В пояснительной записке для каждого типа необходимо указать, какие данные в программе к нему относятся.

2.3.2 Алгоритм решения задачи. Прежде чем приступить к построению схемы алгоритма, предварительно следует определить общую структуру алгоритма решения задачи. Для построения структурированной схемы алгоритма следует воспользоваться технологией пошаговой детализации.

Схема алгоритма (схема программы) должна отображать последовательность операций в программе. При ее изображении используются:

- символы процесса, обозначающие фактические операции обработки данных;
- линейные символы, указывающие направление потока управления;
- специальные символы, предназначенные для упрощения описания и чтения схемы.

При составлении схемы алгоритма решения задачи следует произвести выделение основных этапов в виде подпрограмм. В общем виде любая задача расчета может быть представлена как последовательное выполнение процедур, соответственно реализующих:

- ввод исходных данных и начальных условий;
- выполнение расчета данным численным методом;
- отображение и вывод полученных результатов расчета.

Согласно требованиям структурного программирования, каждый вышеперечисленный этап оформляется в виде отдельной подпрограммы. Тогда блок-схема алгоритма головной (вызывающей) программы состоит из последовательного вызова подпрограмм.

Для каждой подпрограммы составляется схема алгоритма (схема программы). При этом внутри подпрограмм рекомендуется использовать вложенные подпрограммы, которые описывают функционально законченные части вычислительного алгоритма. Схемы алгоритма всех подпрограмм, за исключением стандартных подпрограмм языка программирования, изображаются в курсовом проекте.

При выводе результатов расчета следует обязательно предусмотреть следующие варианты:

- вывод на экран;
- вывод в файл или на печать.

Вывод результатов расчета по программе непосредственно на принтер не рекомендуется, так как при работе в составе локальной сети возможны различного рода сбои и ошибки.

При выводе результатов в файл вначале следует показать значения исходных данных, при которых были получены результаты.

При разработке алгоритма решения задачи следует использовать последовательно для каждой из основных процедур (подпрограмм) следующие методы:

- метод частных целей, который заключается в сведении решения сложной задачи к рассмотрению последовательности более простых задач.
- метод подъема, когда задается начальное самое простое предположение варианта построения алгоритма этапа решения задачи, которое затем последовательно улучшается, пока не достигнет заданного качества.

Эти два метода используются комплексно: сначала сложная задача разбивается на последовательность более простых подзадач, для составления алгоритмов каждой из задач используется метод подъема или опять метод частных целей.

Для упрощения описания блоков схемы, реализующих вычисления переменных по формулам, следует оформлять ссылки на эти формулы.

Схемы алгоритмов составляются на основании требований ГОСТ ИСО/МЭК 2382-1. ЕСПД «Схемы алгоритмов программ, данных и систем. Условные обозначения и правила выполнения».

Если алгоритм программы предусматривает создание дополнительных модулей, то для каждого из них составляется схема алгоритма.

2.4 Реализация программного модуля

2.4.1 Описание структуры разрабатываемой задачи. В данном подразделе определяется структура разрабатываемого пакета с учетом особенности заданной среды программирования.

Рекомендуется не размещать весь программный текст в одном файле программы, а разделить программу по функциональному признаку на несколько отдельных модулей. Например, поместить в отдельный модуль собственные типы данных и подпрограммы реализации заданного метода, а во второй модуль – элементы интерфейса программы. При использовании графического интерфейса в программе в данном разделе определяется число и состав форм. Указываются имена папок и файлов, которые подлежат формированию в составе пакета и их назначение. Допустимо изображение структурной схемы программы, отражающей взаимодействие модулей и форм между собой. Главная форма программы должна раскрываться при запуске программы и содержать следующий минимальный набор элементов:

- заголовок приложения с изображением собственной пиктограммы;
- строки главного меню в верхней части формы;
- строки состояния программы, где отображаются подсказки назначения текущих элементов меню.

2.4.2 Проектирование и реализация интерфейса задачи. Интерфейс программы должен быть графическим с использованием библиотек стандартных элементов API Windows. При описании реализации интерфейса в пояснительной записке приводится обоснование числа и состава форм, описание основных элементов форм, которые будут использоваться в программе. Типовой интерфейс приложения в этом случае должен иметь:

- главную форму программы;
- форму ввода исходных данных с клавиатуры;
- форму отображения результата расчета.

В тексте пояснительной записки следует указать сценарий работы приложения с учетом указанного состава элементов интерфейса.

2.4.3 Разработка справочной системы. Программный продукт должен предоставлять пользователю справочную информацию о назначении и приемах работы. В данном подразделе описывается процесс разработки справочной системы.

2.4.4 Тестирование задачи. В рассматриваемом подразделе производится решение поставленной в задании задачи с использованием указанного в задании математического пакета. Исходные данные для тестирования следует задать самостоятельно таким образом, чтобы результаты выполненного расчета могли быть проверены аналитическим решением или по справочным данным.

Результаты выполнения расчета исходных данных оформляются в пояснительной записке работы.

Тестирование составленного текста программы должно производиться на персональном компьютере. Тестирование программы проводится в три этапа:

- тестирование устойчивости, когда проверяется реакция программы на некорректный ввод данных, значения которых выходят за допустимый диапазон;
- тестирование функциональности, когда выполняется проверка правильности вычисления по известным исходным данным .
- тестирование применимости, где выполняется проверка удовлетворению способов использования требованиям задания (время реакции приложения на команды пользователя, понятность интерфейса, устойчивость вычислительного процесса).

Итоги тестирования программы помещаются в пояснительную записку в виде распечатки файла результатов.

В пояснительной записке следует привести аналитическое подтверждение полученного результата или указать литературный источник, данные которого совпадают с полученным результатом. В приложениях обязательно следует приводить распечатку результатов расчета с указанными значениями исходных данных.

В пояснительной записке следует указать технические характеристики используемого при тестировании компьютера (тип и тактовую частоту центрального процессора, объем и тип оперативной памяти, тип видеоадаптера и монитора, типы и объем внешней памяти) и следующие параметры разработанной программы:

- имена и объем файлов с исходным текстом программы и модулей;

– состав и объем занимаемой внешней памяти всего пакета выполняемых файлов, динамически компонуемых библиотек и дополнительных электронных документов (справочных файлов, графических и файлов мультимедиа, если они использовались);

– требуемые минимальные ресурсы компьютера для использования программы;

– качественная оценка быстродействия программы (например, время счета составляет не более десяти секунд).

2.5 Мероприятия по энерго- и ресурсосбережению

В рассматриваемом подразделе необходимо описать значение мер по энерго- и ресурсосбережению, перечислить мероприятия, привести расчеты потребления электроэнергии компьютером в обычном и «спящем» режимах работы, рассчитать количество сэкономленной электроэнергии.

Для нахождения количества рабочих дней, в течение которых разрабатывался программный продукт, используем формулу

$$n = T_{пк} / (8 - t_{рп}), \quad (1)$$

где $T_{пк}$ – время работы компьютера, ч;

$t_{рп}$ – суммарное время регламентированных перерывов, в течение рабочего дня, ч.

Для нахождения суммарной продолжительности регламентированных перерывов в течение всего времени разработки программного модуля

$$T_{рп} = n \times t_{рп}, \quad (2)$$

где $T_{рп}$ – суммарная продолжительность регламентированных перерывов в течение всего времени разработки программного модуля.

Экономия электроэнергии рассчитывается по формуле

$$C_{э} = T_{рп} \times (W_{пк} - W_{сп}) \times C_{э}, \quad (3)$$

где $W_{пк}$ – потребляемая мощность ПК, кВт;

$W_{сп}$ – потребляемая мощность компьютера в «спящем» режиме;

$W_{пк} = 0,44$ кВт;

$W_{сп} = 0,14$ кВт;

$C_{э}$ – стоимость 1 кВт электроэнергии.

2.6 Заключение

В заключении необходимо описать, для чего предназначен курсовой проект, какие этапы были пройдены при реализации поставленной задачи, проанализировать полученные результаты в ходе выполнения курсового проекта, отразить сильные и слабые стороны разработанной программы, дать рекомендации по ее дальнейшему применению и развитию.

3 Оформление курсового проекта

3.1 Оформление графической части

Графическая часть проекта (входные и выходные формы, алгоритм работы программы) выполняется на двух листах формата А1.

Наименьшая величина высоты отдельного символа процесса и данных схемы - 10 мм, а наименьшее расстояние между блоками – 5 мм, изображение всех блоков структурной схемы выполняют одинаковой высоты и ширины.

В графе «Обозначения основной надписи листа» записывается обозначение согласно требований СТУ 01-32-2017, состоящее из двух букв, сокращенно обозначающих наименование учебного документа, название специальности и семизначного цифрового кода вида: 00.00.000, заканчивающегося обозначением кода документа чертежа Д1, означающего документы прочие.

Пример оформления шифра: КП ПО.00.00.000 Д1.

3.2 Оформление пояснительной записки

Пояснительная записка оформляется на листах писчей бумаги формата А4. Содержание пояснительной записки должно соответствовать требованиям раздела 1.2. Текст пояснительной записки оформляется с соблюдением всех требований СТУ 01-32-2017. Используемые термины должны соответствовать ГОСТ ИСО/МЭК 2382-99. Пояснительная записка должна начинаться с титульного листа. Образец оформления титульного листа курсового проекта находится в приложении А.

Обозначение документа, записываемое в основной надписи листов пояснительной записки, должно совпадать с обозначением основной надписи листа графической части и заканчиваться шифром ПЗ (ПОЯСНИТЕЛЬНАЯ ЗАПИСКА) вместо Д1.

Оформление работы выполняется на персональном компьютере с использованием текстового редактора.

3.2.1 Оформление содержания. Содержание пояснительной записки размещается после бланка задания на курсовую работу.

Образец содержания пояснительной записки находится в приложении Б. Образец заполнения задания для курсового проекта находится в приложении В. Содержание включает наименование разделов и подразделов пояснительной записки с указанием номера страницы. Сквозная нумерация листов пояснительной записки выполняется в правом нижнем углу в рамке, начиная с титульного листа записки (на нем номер не ставится). Содержание фактически расположено на третьей странице пояснительной записки. При оформлении электронного варианта текста пояснительной записки рекомендуется использовать иерархическую структуру заголовков, автоматическую расстановку номеров страниц и вставку автосохранения.

3.2.2 Оформление введения. Введение оформляется на отдельной странице. Объем введения не должен превышать 4 процента от общего состава пояснительной записки.

3.2.3 Вид наиболее часто используемых символов схем алгоритмов программ, а также пример оформления схемы алгоритма, приводятся в приложениях Г и Д методических рекомендаций.

3.2.4 Пример оформления фрагмента программного кода приводится в приложении Е методических рекомендаций.

3.2.5 Оформление и состав заключения. Заключение оформляется на отдельной странице. Объем заключения не должен превышать четыре процента от общего состава пояснительной записки.

3.2.6 Оформление списка используемых источников. Оформление списка используемых источников при выполнении курсового проекта, производится в соответствии с требованиями СТУ 01-32-2017. Примером оформления являются данные методические рекомендации.

Пояснительная записка должна помещаться в обложку для курсового проекта с титульным листом без основной надписи.

Список используемых источников

- 1 ГОСТ ИСО/ МЭК 2382-99. Информационные технологии. Словарь. Ч. 1. Основные термины.
- 2 ГОСТ19.701-90 (ИСО 5807-85). Описание символов. Правила применения символов и выполнения схем.
- 3 СТУ 01-32-2017. Стандарт учреждения. Общие требования к оформлению текстовых документов».
- 4 Архангельский, А.Я. Программирование в среде Delphi 7 / А.Я. Архангельский – Москва: Бином, 2003.
- 5 Баженова, И.Ю. Самоучитель программиста Delphi 7 / И.Ю.Баженова – Москва: КУДИЦ ОБРАЗ, 2003.
- 6 Бобровский, С.И. Delphi 7. Учебный курс/ С.И.Бобровский - Санкт-Петербург: Питер, 2004.
- 7 Васильков, Ю.В., Василькова, Н.Н. Компьютерные технологии вычислений в математическом моделировании / Ю.В.Васильков, Н.Н.Василькова. – Москва: Финансы и статистика, 2004.-255 с.
- 8 Вержбицкий, В.М. Численные методы (линейная алгебра и нелинейные уравнения): Учеб. пособие для вузов / В.М.Вержбицкий. – Москва: Высш.шк., 2000. – 266 с.
- 9 Галисеев, Г.В. Программирование в среде Delphi 7. Самоучитель/ Г.В.Галисеев.-Москва: Издательский дом «Вильямс», 2003.-288 с.
- 10 Фаронов, В.В Программирование баз данных в Delphi 7 / В.В. Фаронов – Санкт-Петербург: Питер, 2006.
- 11 Шпак, Ю.А. Delphi 7 на примерах / Ю.А.Шпак – Украина: Юниор, 2003.

Приложение А
(обязательное)

Пример оформления титульного листа

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
ГЛАВНОЕ УПРАВЛЕНИЕ ПО ОБРАЗОВАНИЮ
МОГИЛЕВСКОГО ОБЛАСТНОГО ИСПОЛНИТЕЛЬНОГО КОМИТЕТА
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«МОГИЛЕВСКИЙ ГОСУДАРСТВЕННЫЙ ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ»

Специальность	2-40 01 01
Учебная группа	ПО-405
Учебная дисциплина	Основы алгоритмизации и программирования

КУРСОВОЙ ПРОЕКТ

**РАЗРАБОТКА ПРОГРАММЫ «СПРАВОЧНИК ПО ОСНОВНЫМ
КОМПОНЕНТАМ DELPHI»**

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

КП ПО.00.00.000 ПЗ

Разработал	Е.Д.Медведев
Руководитель проекта	Ю.А.Кашпарова

Приложение Б
(обязательное)

Пример оформления содержания

Содержание

Введение	5
1 Общая характеристика задачи	6
2 Проектирование программного модуля	7
2.1 Описание входных и выходных данных	7
2.2 Алгоритм решения задачи	9
3 Реализация программного модуля	11
3.1 Описание структуры разрабатываемой задачи	11
3.2 Проектирование и реализация интерфейса задачи	12
3.3 Разработка справочной системы	13
3.4 Тестирование задачи	14
4 Мероприятия по энерго- и ресурсосбережению	17
Заключение	18
Список используемых источников	19
Приложение А (обязательное) Текст программы	20

					КП ПО.00.00.000 ПЗ				
Изм.	Лист	№ докум.	Подп.	Дата	Разработка программы «Справочник по основным компонентам Delphi»	Лит.	Лист	Листов	
Разраб.		Медведев							
Пров.		Кашпарова							
Н. контр.									
Утв.					Пояснительная записка	МГПК гр. ПО-405			

Приложение В
(обязательное)

Пример оформления бланка задания

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
ГЛАВНОЕ УПРАВЛЕНИЕ ПО ОБРАЗОВАНИЮ
МОГИЛЕВСКОГО ОБЛАСТНОГО ИСПОЛНИТЕЛЬНОГО КОМИТЕТА
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«МОГИЛЁВСКИЙ ГОСУДАРСТВЕННЫЙ ПОЛИТЕХНИЧЕСКИЙ КОЛЛЕДЖ»

УТВЕРЖДАЮ
Председатель цикловой комиссии

подпись

дата

ЗАДАНИЕ
на курсовой проект

по учебной дисциплине _____ Основы алгоритмизации и программирования
учащемуся _____

(фамилия, имя, отчество)

курса _____ учебной группы _____

Тема курсового проекта _____ Разработка программы «Справочник по основным
компонентам Delphi» _____

Исходные данные _____

Состав проекта:

Пояснительная записка

Наименование раздела пояснительной записки	Сроки выполнения
Введение	
1 Общая характеристика задачи	
2 Проектирование программного модуля	
2.1 Описание входных и выходных данных	
2.2 Алгоритм решения задачи	
3 Реализация программного модуля	
3.1 Описание структуры разрабатываемой задачи	
3.2 Проектирование и реализация интерфейса задачи	
3.3 Разработка справочной системы	
3.4 Тестирование задачи	
4 Мероприятия по энерго- и ресурсосбережению	
Заключение	

Наименование раздела пояснительной записки	Сроки выполнения
Список используемых источников	
Приложение А (обязательное) Текст программы	

Графическая часть

Лист – 1 Входные и выходные формы	
Лист – 2 Алгоритм работы программы	

Дата выдачи _____

Срок окончания _____

Руководитель _____ Кашпарова Ю.А.

Подпись учащегося _____

Приложение Г (информационное)

Основные символы схемы алгоритма программы



Рисунок Г.1 – Изображение символа «Данные»

Символ отображает данные, носитель данных не определен. Используется для описания операций ввода-вывода данных. Внутри символа записываются операция (чтение или запись) и идентификаторы переменных, участвующих в данной операции. Источник данных (клавиатура, файл или экран) рекомендуется указывать с помощью символа комментария.

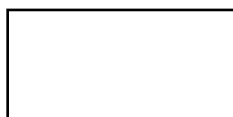


Рисунок Г.2 – Символ «Процесс»

Символ отображает операцию или группу операций обработки данных любого вида. Внутри символа записывается выражение, используемое при обработке данных. С его помощью также может быть описана практически любая обобщенная операция, если внутри символа описать ее действия.



Рисунок Г.3 – Символ «Предопределенный процесс» (подпрограмма)

Символ отображает предопределенный процесс, состоящий из операций, которые определены в другом месте (в подпрограмме, модуле). Используется для указания вызова подпрограммы процедуры (или функции, не возвращающей результат). Внутри символа указывается имя подпрограммы со списком параметров.

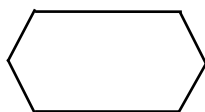


Рисунок Г.4 – Символ «Подготовка» (модификация)

Символ отображает модификацию команды или группы команд с целью воздействия на некоторую предыдущую функцию (установка переключателя, модификация индексного регистра или инициализация программы). Используется для обозначения цикла `for`. Внутри записывается диапазон изменения индексной переменной.

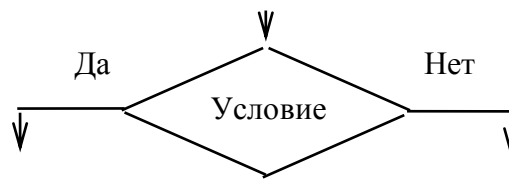


Рисунок Г.5 – Символ “Решение”

Символ отображает условное ветвление с помощью оператора `if` или функцию переключательного типа, имеющую один вход и ряд альтернативных выходов, только один из которых может быть активирован после вычисления условия. Внутри символа записывается условие.

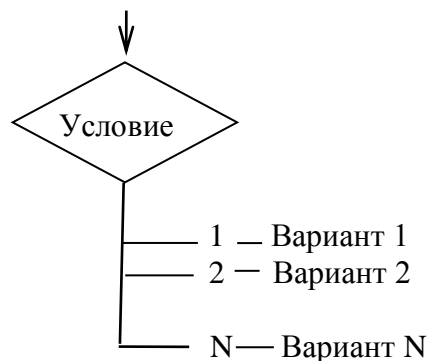


Рисунок Г.6 - Изображение структуры с выбором из N вариантов

Если при ветвлении число альтернатив более двух (например, для операторов выбора или переключателей), то каждый выход символа решения должен сопровождаться соответствующими значениями условий для изображения логического пути вычислительного процесса.

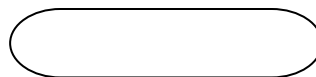


Рисунок Г.7 – Специальный символ “Терминатор”

Символ отображает выход во внешнюю среду или вход из внешней среды (начало и конец программы, внешнее использование и источник или пункт назначения данных). Символ имеет только один вход или выход линий потока управления.



Рисунок Г.8 – Специальный символ “Соединитель”

Символ отображает выход в часть схемы и вход из другой части схемы и используется для обрыва линии связи и продолжения ее в другом месте. Внутри соответствующих разъединителей ставится одинаковый уникальный идентификатор. При оформлении межстраничного переноса используется с символом комментария, внутри которого указывается на какую страницу (или из какой) передается управление.

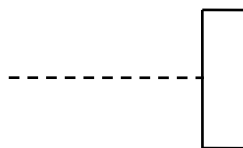


Рисунок Г.9 – Специальный символ «Комментарий»

Используется для добавления описательных комментариев или пояснительных записей в целях объяснения или примечаний. Текст комментария оформляется справа от символа.



Рисунок Г.10 - Специальный символ «Пропуск»

Символ (три точки) используется в символах линий для пропуска одного или группы символов, в которой не определены ни тип, ни число символов. Используется для отображения общих решений с неизвестным числом повторений.

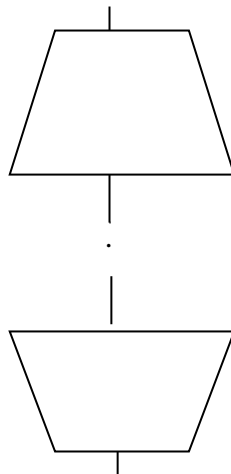


Рисунок Г.11 - Изображение символов «Границы цикла»

Символ отображает границы циклов с предусловием или с постусловием. Обе части имеют одинаковый идентификатор. Внутри символов размещаются условия для инициализации, приращения завершения в начале или в конце в зависимости от расположения операции, проверяющей условие. Между символами записываются символы процесса, обозначающие операции, которые выполняются в теле цикла.

Приложение Д (информационное)

Пример схемы алгоритма



Рисунок Д.1 – Пример схемы алгоритма головной программы

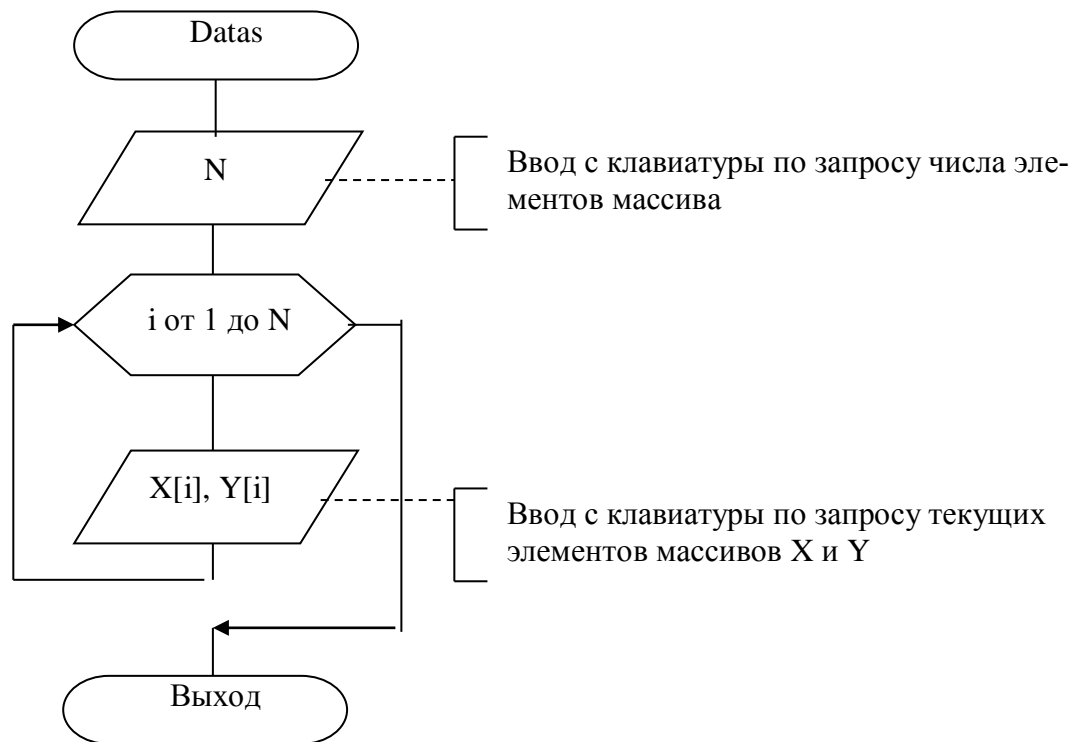


Рисунок Д.2 – Пример схемы алгоритма подпрограммы ввода двух одномерных матриц

Приложение Е (информационное)

Пример оформления фрагмента программного кода

Написание кода для ввода исходных данных

Для ввода исходных данных используются компоненты Edit, находящиеся на главной форме. Компоненты, используемые для ввода данных, имеют строковый тип, а для расчетов необходим числовой тип. Для перевода строковых переменных в числовые используются функции `strtoint`, `strtofloat`, а для перевода числовых переменных в строковые используются функции `inttostr`, `floattostr`:

```
x1:= strtofloat(Edit3.Text);  
n:= strtoint(Edit5.Text;,  
stringgrid1.Cells[1,1]:=floattostr(x).
```

В программе имеется защита от некорректного ввода, которая не позволяет пользователю ввести некорректные данные:

```
if key in ['0'..'9','#8','-','.','] then  
else  
key:=#0.
```