

Министерство образования и науки Российской Федерации
Национальный исследовательский ядерный университет «МИФИ»
Балаковский инженерно-технологический институт –
филиал НИЯУ МИФИ

ПРОГРАММИРОВАНИЕ И ОСНОВЫ АЛГОРИТМИЗАЦИИ

Методические указания к выполнению курсовой работы
для студентов направления «Управление в технических системах»
всех форм обучения

*Одобрено
редакционно-издательским советом
Балаковского инженерно-технологического
института*

Балаково 2016

ВВЕДЕНИЕ

Выполнение курсовой работы по дисциплине «Программирование и основы алгоритмизации» является важной составной и обязательной частью самостоятельной работы студентов по направлению «Управление в технических системах». Оно имеет своей целью: расширение и закрепление теоретических и прикладных знаний, умелое использование их при решении конкретных задач.

С учетом рекомендуемой тематики студентам предоставляется право выбора темы курсовой работы. Они могут предложить и свою тему, обосновав целесообразность и необходимость ее выполнения.

Выполнение курсовой работы необходимо начинать с изучения специальной литературы по избранной теме. При работе с литературными источниками студент дополнительно подбирает научную литературу. Во время работы над курсовой работой можно использовать INTERNET- ресурсы.

После того, как тема выбрана и согласована с руководителем курсовой работы, составляется календарный план, в котором определяются сроки выполнения этапов курсовой работы (сбор материала, обработка собранного и изученного материала, разработка приложения, его отладка и тестирование, оформление работы т.д.) Составление календарных планов облегчает проведение контроля за выполнением исследования и помогает студенту самостоятельно и осознано выполнять курсовую работу. План работы согласовывается с руководителем. Логические контуры плана оформляются определенной структурой курсовой работы.

Целью выполнения курсовой работы по дисциплине «Программирование и основы алгоритмизации» является формирование у студентов опыта создания приложений в среде программирования C++.

К задачам выполнения курсовой работы относятся:

- закрепление, углубление, расширение и систематизация знаний, полученных при изучении дисциплин «Информатика», «Программирование и

основы алгоритмизации», а также приобретение практических навыков создания приложений;

- приобретение опыта аналитической и расчетной работы, развитие умений работы со специальной литературой и иными информационными источниками;
- формирование умений формулировать логически обоснованные выводы, предложения и рекомендации по результатам выполненной работы;
- формирование умения грамотно подготовить презентацию защищаемой работы;
- формирование умения выступать перед аудиторией с докладом при защите курсовой работы, компетентно отвечать на вопросы, вести профессиональную дискуссию, убеждать оппонентов в правильности принятых решений;
- развитие у студента системного мышления, а также профессиональной устной и письменной речи.

В результате выполнения курсовой работы студент должен научиться:

- создавать программу в среде программирования C++ в соответствии с основными этапами ее разработки;
- строить блок-схему алгоритма работы программы в соответствии с требованиями ГОСТ 19.701-90;
- грамотно тестировать программу;
- анализировать результаты работы программы и делать выводы.

ТРЕБОВАНИЯ К ОФОРМЛЕНИЮ КУРСОВОЙ РАБОТЫ

Курсовая работа должна отражать высокий уровень профессиональной подготовки, должна быть написана четким, экономически грамотным языком и правильно оформлена.

При написании курсовой работы необходимо выполнять следующие требования по оформлению работы:

1. Пояснительная записка выполняется на листах со штампом. Шифр в штампе выбирается согласно теме работы в соответствии с классификатором ЕСКД.

2. Заголовок главы пишется с прописными буквы с красной строки и с выравниванием по ширине. Каждая глава курсовой работы начинается с нового листа. Пункты главы, если они есть, имеют порядковую нумерацию и нумеруются арабскими цифрами, состоящими из номера главы и номера пункта в пределах главы, например, 1.1, 1.2, 2.1, 2.2 и т.д. Заголовки глав или пунктов отделяются от основного текста пустой строкой. Между заголовков главы и пункта пустая строка не оставляется.

3. Страницы курсовой работы нумеруются арабскими цифрами в правом нижнем углу штампа. Титульный лист включается в общую нумерацию курсовой работы, но номер страницы на нем не проставляется.

4. Шрифт текста Times New Roman, размер – 14, междустрочный интервал – полуторный, абзацы формируются с отступом 1,25 см.

5. Все иллюстрации, кроме таблиц, являются рисунками и выравниваются по центру. Наименование иллюстрации помещается под ней с красной строки, нумерация иллюстрации является сквозной для всей курсовой работы или отдельно по главам. Пример подписи иллюстрации:

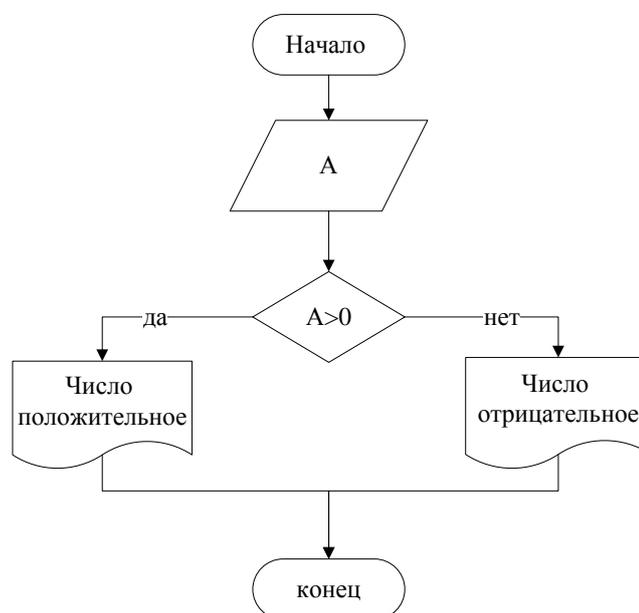


Рисунок 1- Блок-схема проверки знака числа

6. Таблицы нумеруются арабскими цифрами сквозной нумерацией. Заголовок таблиц размещается над таблицей слева. Пример оформления заголовка таблицы:

Таблица 1- Наименование

7. Формулы нумеруются арабскими цифрами сквозной нумерацией для всей курсовой работы либо по главам, например:

$$y=kx+b, \tag{1}$$

где k – коэффициент пропорциональности;

b – смещение, относительно оси абсцисс.

8. Список литературы оформляется в соответствии с ГОСТ 7.1-2003. Источники в списке литературы должны быть указаны в алфавитном порядке в следующей последовательности:

- 1) нормативно-инструктивные материалы;
- 2) учебники, монографии, другие книги;
- 3) журнальные статьи;
- 4) газетные статьи.

Список использованных для выполнения теоретической части курсовой работы источников должен включать не менее 10 наименований.

При использовании литературы необходимо обращать внимание на дату издания - желательно использовать литературу, не старше 5 лет.

СОДЕРЖАНИЕ КУРСОВОЙ РАБОТЫ

Курсовая работа имеет следующую структуру:

Титульный лист (прил. 1);

Содержание, с указанием страниц разделов;

Введение

1 Техническое задание

2 Схемы алгоритмов

2.1 Схема алгоритма переключателя

.....

3. Текст программы

3.1 Текст программы формы «Приветствие»

...

4 Тестовые примеры для всех ветвей работы программы

5 Руководство пользователя

Заключение

Список использованной литературы;

Приложения.

1. Введение

Во введении обосновываются актуальность выбранной темы, степень ее разработанности, цель и задачи, поставленные при выполнении курсовой работы, на каком материале выполнена работа. Введение пишется на 1-2 страницах.

Заключение содержит выводы по курсовой работе

По результатам выполнения курсовой работы необходимо сделать выводы, которые должны отражать следующие вопросы:

- 1) Какова была задача выполнения работы? Была ли она решена и достигнуты ли основные цели?
- 2) Какие новые знания и навыки вы получили в процессе выполнения работы?
- 3) Где могут быть применены полученные результаты?

2. Техническое задание

должно содержать следующие разделы:

1. введение;
2. наименование и область применения;
3. основание для разработки;
4. назначение разработки;
5. технические требования к программе или программному изделию;
6. технико-экономические показатели;
7. стадии и этапы разработки;
8. порядок контроля и приемки;

Содержание разделов технического задания:

2.1. Введение должно включать краткую характеристику области применения программы, а также объекта, в котором предполагается её использовать. Основное назначение введения — продемонстрировать актуальность данной разработки и показать, какое место эта разработка занимает в ряду подобных.

2.2. В разделе «Наименование и область применения» указывают наименование, краткую характеристику области применения программы или программного изделия и объекта, в котором используют программу или программное изделие.

2.3. В разделе «Основание для разработки» должны быть указаны:

- документ (документы), на основании которых ведется разработка. Таким документом может служить план, приказ, договор и т. п.;
- организация, утвердившая этот документ, и дата его утверждения;
- наименование и (или) условное обозначение темы разработки.

2.4. В разделе «Назначение разработки» должно быть указано функциональное и эксплуатационное назначение программы или программного изделия.

2.5. Раздел «Технические требования к программе или программному изделию» должен содержать следующие подразделы:

- требования к функциональным характеристикам;
- требования к надежности;
- условия эксплуатации;
- требования к составу и параметрам технических средств;
- требования к информационной и программной совместимости;
- требования к маркировке и упаковке;
- требования к транспортированию и хранению;
- специальные требования.

2.5.1. В подразделе «Требования к функциональным характеристикам» должны быть указаны требования к составу выполняемых функций, организации входных и выходных данных, временным характеристикам и т. п.

2.5.2. В подразделе «Требования к надежности» должны быть указаны требования к обеспечению надежного функционирования (обеспечение устойчивого функционирования, контроль входной и выходной информации, время восстановления после отказа и т. п.).

2.5.3.В подразделе «Условия эксплуатации» должны быть указаны условия эксплуатации (температура окружающего воздуха, относительная влажность и т. п. для выбранных типов носителей данных), при которых должны обеспечиваться заданные характеристики, а также вид обслуживания, необходимое количество и квалификация персонала.

2.5.4.В подразделе «Требования к составу и параметрам технических средств» указывают необходимый состав технических средств с указанием их технических характеристик.

2.5.5.В подразделе «Требования к информационной и программной совместимости» должны быть указаны требования к информационным структурам на входе и выходе и методам решения, исходным кодам, языкам программирования. При необходимости должна обеспечиваться защита информации и программ.

2.5.6.В подразделе «Требования к маркировке и упаковке» в общем случае указывают требования к маркировке программного изделия, варианты и способы упаковки.

2.5.7. В подразделе «Требования к транспортированию и хранению» должны быть указаны для программного изделия условия транспортирования, места хранения, условия хранения, условия складирования, сроки хранения в различных условиях

2.5.8. В разделе «Технико-экономические показатели» должны быть указаны: ориентировочная экономическая эффективность, предполагаемая годовая потребность, экономические преимущества разработки по сравнению с лучшими отечественными и зарубежными образцами или аналогами.

2.6.В разделе «Стадии и этапы разработки» устанавливают необходимые стадии разработки, этапы и содержание работ (перечень программных документов, которые должны быть разработаны, согласованы и утверждены), а также, как правило, сроки разработки и определяют исполнителей.

2.7.В разделе «Порядок контроля и приемки» должны быть указаны виды испытаний и общие требования к приемке работы.

2.8.В приложениях к техническому заданию при необходимости приводят:

- перечень научно-исследовательских и других работ, обосновывающих разработку;

- схемы алгоритмов, таблицы, описания, обоснования, расчеты и другие документы, которые могут быть использованы при разработке;

- другие источники разработки.

В случаях, если какие-либо требования, предусмотренные техническим заданием, заказчик не предъявляет, следует в соответствующем месте указать «Требования не предъявляются».

Пример технического задания приведен в приложении 2.

3. Блок-схема решения задачи

Для разработки структуры программы удобнее пользоваться записью алгоритма в виде блок-схемы. Блок-схема работы программы должна строиться в соответствии с требованиями ГОСТ 19.701 – 90 (ИСО 5807 – 85) Схемы алгоритмов, программ, данных и систем.

В качестве инструментария для построения блок-схем могут быть выбраны: MS Word, MS Visio, а также специализированные редакторы блок-схем.

4 Исходный текст программы

Текст программы обязательно должен содержать подробные комментарии. Комментарии должны пояснять не менее 70% текста программы.

Важным моментом является разработка дружественного пользовательского интерфейса.

Основные требования к дизайну пользовательского интерфейса:

Общий дизайн:

1. Оставляйте свободное место между краями формы и ее элементами управления

2. Располагайте информацию, выводимую приложением, под элементами управления, обеспечивающими ввод, и/или справа от них

3. Когда вы перетаскиваете элементы управления, среда разработки выводит синие и фиолетовые линии, способствующие выравниванию элементов управления.

Синие линии помогают выровнять элементы относительно друг друга, а фиолетовые помогают расположить элементы управления относительно текста

4. При задании имени элементов используйте префиксы: Form – для формы, Label – для меток, TextBox – для текстовых полей, Button – для кнопок, PictureBox – для графического блока.

Формы:

1. Начинайте заголовок формы с прописной буквы, а в конце не ставьте никакие знаки пунктуации.

2. Устанавливайте у формы шрифт 9 pt Segoe UI

Кнопки:

1. Кнопки располагаются вертикально вдоль правого края формы (начиная с самого верха) или идут вдоль нижнего края справа налево, начиная от правого края.

Метки:

1. Метки, используемые для описания элементов управления, должны начинаться с прописной буквы и заканчиваться двоеточием. Такие метки называются описательными.

2. Свойство TextAlign (Выравнивание текста) описательной метки должно иметь значение MiddleLeft (Посередине слева). Это обеспечит выравнивание текста в пределах группы меток.

3. Располагайте описательную метку либо над идентифицируемым элементом управления (например, текстовым полем), либо слева от него.

4. Выравнивайте описательные метки в группе по левому или правому краю, если метки расположены вертикально.

5. Для идентификации метки вывода используйте описательную метку.

6. Метки вывода должны отличаться от описательных меток. Этого можно достичь, установив свойство BorderStyle (Стиль рамки) в значение Fixed3D

7. Если несколько меток вывода расположены друг под другом и отображают числа, используемые в математических вычислениях устанавливайте свойство TextAlign (Выравнивание текста) в значение MiddleRight

8. Описательная метка и элемент управления, который она идентифицирует, должны быть выровнены левыми краями, если они расположены вертикально.

9. Текст в описательной метке и текст в элементе управления, который она идентифицирует, должны быть выровнены по нижнему краю, если метка и элемент расположены горизонтально.

Графические блоки:

1. Используйте элемент PictureBox (Графический блок) для улучшения внешнего вида графического интерфейса с помощью изображений, которые пользователь не может изменить.

2. Изображения должны по размеру соответствовать своим графическим блокам. Это достигается установкой свойстваSizeMode (Режим определения размера) в значение StretchImage (Растянуть изображение)

Текстовые поля:

1. Для ввода данных с клавиатуры пользуйтесь текстовыми полями.

2. У каждого текстового поля должна быть описательная метка, разъясняющая, какие данные ожидаются от пользователя.

3. Устанавливайте у текстовых полей ширину, достаточную для предполагаемых вводимых данных.

4. Когда вы располагаете текстовые поля друг под другом, выравнивайте их по правому краю и, если возможно, делайте их одинакового размера. Описательные метки этих полей выравнивайте по левому краю.

В данных методических указаниях приведены только некоторые правила и рекомендации по дизайну приложения.

Результаты тестирования должны быть представлены в разделе «Тестовые примеры».

4. Тестовые примеры для всех ветвей работы программы

В любой программе должна быть предусмотрена защита от некорректных действий пользователя, а также обработка иных ошибок. Иначе, в случае задания пользователем некорректных параметров или отсутствия файла графического модуля в директории, используемой по умолчанию, или при возникновении иных

исключительных ситуаций, программа будет «вылетать». Обработка ошибок предполагает выдачу пользователю рекомендаций по их устранению.

В данном разделе необходимо протестировать все ветви работы программы на наличие возможных ошибок:

1. Ввод данных (попытаться определить все ошибки, которые возникнут при вводе исходных данных пользователем).

2. Решение.

3. Вывод результатов.

Результаты тестирования нужно проиллюстрировать скриншотами, показывающими, как будет реагировать программа на возникновение той или иной ошибки.

5. Руководство пользователя

Одним из этапов создания программного обеспечения является разработка руководства по работе с программой – руководство пользователя.

Оно должно содержать подробные инструкции по работе с программой, а также экранные формы, иллюстрирующие текст.

Руководство пользователя должно быть организовано таким образом, чтобы любому человеку, не знакомому со средой программирования, программой и методом, реализуемым ею, были понятны действия, которые от него требуется выполнить.

Примерный план написания руководства пользователя:

1. Действия при запуске программы (какой файл должен быть запущен, какие действия должны при этом выполняться).

2. Структура программы (какие окна есть, как между ними переключаться, структура меню и назначение его элементов).

3. Каким образом осуществляется ввод данных.

4. Как запустить программу на выполнение.

5. Какая информация выводится на экран.

Таким образом, нужно рассмотреть последовательность всех операций по работе с программой на конкретном примере, делая подробные комментарии и иллюстрируя текст графически.

В случае необходимости можно указать следующую информацию: требуются ли от пользователя какие-либо настройки, если да, то какие, нужно ли использовать дополнительное программное обеспечение и т.п.

Руководство пользователя, как правило, содержит следующие разделы:

1. общие сведения о программном продукте;
2. описание установки;
3. описание запуска;
4. инструкции по работе (или описание пользовательского интерфейса);
5. сообщения пользователю.

Раздел Общие сведения о программе обычно содержит наименование программного продукта, краткое описание его функций, реализованных методов и возможных областей применения.

Раздел Установка обычно содержит подробное описание действий по установке программного продукта и сообщений, которые при этом могут быть получены.

В разделе Запуск, как правило, описаны действия по запуску программного продукта и сообщений, которые при этом могут быть получены.

Раздел Инструкции по работе обычно содержит описание режимов работы, форматов ввода-вывода информации и возможных настроек.

Раздел Сообщения пользователю должен содержать перечень возможных сообщений, описание их содержания и действий, которые необходимо предпринять по этим сообщениям.

РЕЦЕНЗИЯ НА КУРСОВУЮ РАБОТУ И ДОПУСК К СОБЕСЕДОВАНИЮ

После написания и соответствующего оформления курсовая работа сдается для рецензирования не позднее, чем за неделю до собеседования. После проверки преподавателем студент получает работу с рецензией, в которой указываются

положительные стороны и недостатки работы, делается вывод о допуске (не допуске) к собеседованию.

При отрицательной рецензии студент, руководствуясь замечаниями преподавателя, должен доработать курсовую работу и снова представить на кафедру "ИУС".

К собеседованию допускаются студенты, имеющие работы с положительной рецензией. Если студент получил отрицательную рецензию или если рекомендовано доработать какой-либо раздел работы, то это должно быть сделано до собеседования и оформлено в виде вставок в соответствующие разделы. Эти дополнения должны быть использованы при устном ответе в ходе собеседования по работе.

ТЕМАТИКА КУРСОВЫХ РАБОТ

1. Разработка тестирующих приложений
2. Разработка приложений, имитирующих игры
3. Разработка приложений обработки баз данных
4. Разработка графических приложений
5. Разработка Web-приложений

ЛИТЕРАТУРА

1. ГОСТ 7.1 – 2003. Библиографическое описание документа. Общие требования и правила составления [текст] – взамен ГОСТ 7.1-84, ГОСТ 7.16-79, ГОСТ 7.18-79, ГОСТ 7.34-81, ГОСТ 7.40-82 – введ. 2004 – 07 – 01. – М.: Издательство стандартов, 2004. – 141с. – (Система стандартов по информации, библиотечному и издательскому делу).
2. ГОСТ 7.82 – 2001. Библиографическая запись. Библиографическое описание электронных ресурсов. Общие требования и правила составления [текст] – введ. 2002 – 07 – 01 – М.: Издательство стандартов, 2001. – 35с. – (Система стандартов по информации, библиотечному и издательскому делу).

3. ГОСТ 19.701 – 90 (ИСО 5807 – 85) Схемы алгоритмов, программ, данных и систем [текст]. – взамен ГОСТ 19.002-80, ГОСТ 19.003-80 – введ. 1992 – 01 – 01. – М.: Государственный стандарт союза ССР, 1990. – 22с.
4. Павловская Т. А. С/С++. Программирование на языке высокого уровня: Учебник для вузов. Санкт-Петербург: Питер, 2013 г. , 464 с.
5. Павловская Т.А.С/С++. Процедурное и объектно-ориентированное программирование. Учебник для вузов. Санкт-Петербург: Питер, 2015 г. , 496 с.

СОДЕРЖАНИЕ

Введение	2
Требования к оформлению курсовой работы.....	3
Содержание курсовой работы	5
Рецензия на курсовую работу и допуск к собеседованию	13
Тематика курсовых работ	14
Литература	14
Приложение 1.....	16
Приложение 2.....	17

Балаковский инженерно-технологический институт — филиал
федерального государственного автономного образовательного учреждения
высшего образования
«Национальный исследовательский ядерный университет «МИФИ»

Факультет Атомной энергетики и технологий

Направление Управление в технических системах

Кафедра «Информатика и управление в технических системах»

КУРСОВАЯ РАБОТА

по дисциплине

Основы алгоритмизации и программирования

«ТЕМА»

Выполнил студент группы
УПТС-11

Допущен к защите

Руководитель КР

Фролова М.А. _____

« ____ » _____ 201_ г.

Защитил с оценкой

Фролова М.А. _____

« ____ » _____ 201_ г.

Балаково 201_

Балаковский инженерно-технологический институт — филиал
федерального государственного автономного образовательного учреждения
высшего образования
«Национальный исследовательский ядерный университет «МИФИ»

Факультет Атомной энергетики и технологий

Направление Управление в технических системах

Кафедра «Информатика и управление в технических системах»

УТВЕРЖДАЮ
Зав. кафедрой ИУС
к.т.н., доц. Фролова М.А.
« _____ » _____ 20__ г.

Программа сортировки одномерного массива

Техническое задание на разработку приложения

Листов 3

Руководитель,
к.т.н., доцент _____ Фролова М.А.
Исполнитель,
студент 1 курса направления УПТС
_____ Ф.И.О.

Балаково 201_

1. Введение

Настоящее техническое задание распространяется на разработку программы сортировки одномерного массива методами пузырька, прямого выбора, Шелла и быстрой сортировки, предназначенной для использования школьниками старших классов при изучении курса школьной информатики.

2. Основание для разработки

2.1. Программа разрабатывается на основе учебного плана направления «Управление в технических системах».

2.2. Наименование работы: «Программа сортировки одномерного массива».

2.3. Исполнитель: студент УПТС- 11ФМО.

2.4. Соисполнители: нет.

3. Назначение

Программа предназначена для использования школьниками при изучении темы «Обработка одномерных массивов» в курсе «Информатика».

4. Требования к программе или программному изделию

4.1. Требования к функциональным характеристикам

4.1.1. Программа должна обеспечивать возможность выполнения следующих функций:

- ввод размера массива и самого массива;
- хранение массива в памяти;
- выбор метода сортировки;
- вывод текстового описания метода сортировки;
- вывод результата сортировки.

4.1.2. Исходные данные:

- размер массива, заданный целым числом;
- массив.

4.1.3. Организация входных и выходных данных

Входные данные поступают с клавиатуры.

Выходные данные отображаются на экране и при необходимости выводятся на печать

4.2. Требования к надежности

Предусмотреть контроль вводимой информации.

Предусмотреть блокировку некорректных действий пользователя при работе с системой.

4.3. Условия эксплуатации.

Стандартные условия эксплуатации программных продуктов. Необходимые сотрудники для обслуживания серверной части системы – системный администратор для обслуживания собственно сервера (регистрация и удаление пользователей, добавление и настройка учебных материалов) и группа разработчиков учебных курсов, численность и состав которой зависит от конкретной дисциплины курса

4.4. Требования к составу и параметрам технических средств.

Система должна работать на IBM-совместимых персональных компьютерах.

Минимальная конфигурация:

тип процессора	Pentium I и выше
объем оперативного запоминающего устройства	32 Мб и более
объем свободного места на жестком диске	40 Мб

Рекомендуемая конфигурация:

тип процессора	Pentium II 400
объем оперативного запоминающего устройства	128 Мб
объем свободного места на жестком диске	60 Мб

4.5. Требования к программной совместимости.

Программа должна работать под управлением семейства операционных систем Win 32 (Windows 10/8/7/XP и т. п.).

4.6 Требования к маркировке и упаковке.

Не предъявляются.

4.7 Требования к транспортированию и хранению.

Не предъявляются.

4.8 Специальные требования.

Не предъявляются.

5. Требования к программной документации

5.1. Разрабатываемые программные модули должны быть самодокументированы, т. е. тексты программ должны содержать все необходимые комментарии.

5.2. Разрабатываемая программа должна включать справочную информацию о работе программы, описания методов сортировки и подсказки учащимся.

5.3. В состав сопровождающей документации должны входить:

5.3.1. Пояснительная записка на пяти листах, содержащая описание разработки.

5.3.2. Руководство пользователя.

ПРОГРАММИРОВАНИЕ И ОСНОВЫ АЛГОРИТМИЗАЦИИ

Методические указания к выполнению курсовой работы
для студентов направления «Управление в технических системах» всех форм
обучения

Составила Фролова Марина Александровна