МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
ДОНЕЦКОЙ НАРОДНОЙ РЕСПУБЛИКИ
ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

УТВЕРЖДАЮ:
Первый проректор
_____ А.А. Каракозов
(подпись)
«31» марта 2023 года

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.О.21 Программирование
(код и наименование дисциплины согласно учебному плану)

Направление подготовки (специальность): 09.03.01 Информатика и вычислительная техника
(код и наименование направления подготовки / специальности)

Направленность (профиль): Системное программирование
(наименование профиля / магистерской программы /специализации)

Программа: бакалавриат
(бакалавриат, магистратура, специалитет)

Форма обучения: очная
(очная, заочная, очно-заочная)

Форма обучения:	Очная
Семестр(ы)	1, 2
Общая трудоёмкость в з.е./часах	10/360
Контактная работа (час.), в том числе:	180 (89+91)
Лекции (час.)	68 (34+34)
Лабораторные работы (час.)	102 (51+51)
Практические (семинарские) занятия (час.)	0
Самостоятельная работа (час.), в том числе	90 (55+35)
Курсовой проект(работа) (семестр/час.)	2/27 (0+27)
Контроль (экзамен, час./зачёт)	2×экзам.,90 (36+54)час.

Донецк, 2023 г.

Рабочая программа дисциплины «Программирование» составлена в соответствии с учебным планом по направлению подготовки (специальности) 09.03.01 «Информатика и вычислительная техника», профиль «Системное программирование» для 2023 года приёма по очной форме обучения.

Составитель:
доцент кафедры
компьютерной инженерии,
к.т.н. _____ Дорожко Л.И.

Рабочая программа **рассмотрена и принята** на заседании кафедры «Компьютерная инженерия».
Протокол от « 21 » марта 2023 года № 8

Заведующий кафедрой _____ Аноприенко А.Я.
(подпись)

Рабочая программа **одобрена учебно-методической комиссией** ГОУВПО «ДонНТУ» по направлению подготовки 09.03.01 «Информатика и вычислительная техника»
Протокол от « 21 » марта 2023 года № 4

Председатель _____ Аноприенко А.Я.
(подпись)

Рабочая программа **продлена** для 20___ года приёма на заседании кафедры «Компьютерная инженерия».
Протокол от «_____» _____20___ года № _____
Заведующий кафедрой _____ Аноприенко А.Я.
(подпись)

Рабочая программа **продлена** для 20___ года приёма на заседании кафедры «Компьютерная инженерия».
Протокол от «_____» _____20___ года № _____
Заведующий кафедрой _____ Аноприенко А.Я.
(подпись)

Рабочая программа **продлена** для 20___ года приёма на заседании кафедры «Компьютерная инженерия».
Протокол от «_____» _____20___ года № _____
Заведующий кафедрой _____ Аноприенко А.Я.
(подпись)

1. ОБЪЕКТ, ЦЕЛЬ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Цели дисциплины - изучение основных понятий и методов разработки алгоритмов и их реализации на конкретном алгоритмическом языке. Основными задачами изучения дисциплины "Программирование" научить студентов разрабатывать алгоритмы задач и реализовать их в программах на конкретном языке.

В результате освоения дисциплины студент должен:

- *знать*

основы математики, физики, вычислительной техники и программирования;

основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий;

методики использования программных средств для решения практических задач;

- *уметь*

решать стандартные профессиональные задачи с применением естественнонаучных и общеинженерных знаний, методов математического анализа и моделирования;

применять языки программирования, современные программные среды разработки информационных систем, решения прикладных задач различных классов;

использовать программные средства для решения практических задач;

- *владеть навыками*

теоретического и экспериментального исследования объектов профессиональной деятельности;

программирования, отладки и тестирования прототипов программно-технических комплексов;

использования программных средств для решения практических задач.

Процесс изучения дисциплины направлен на формирование следующих компетенций:

- способности применять естественнонаучные и общеинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности (ОПК-1);
- способности разрабатывать алгоритмы и программы, пригодные для практического применения (ОПК-8);
- способности осваивать методики использования программных средств для решения практических задач (ОПК-9).

2. МЕСТО ДИСЦИПЛИНЫ В ОСНОВНОЙ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЕ

Дисциплина относится к обязательной части Блока 1 «Дисциплины

(модули)» учебного плана.

Знания и умения, приобретенные при освоении данной дисциплины, реализуются студентом при изучении последующих дисциплин: «Системное программное обеспечение», «Системное программирование», «Организация баз данных», «Программирование в среде современных ОС» при прохождении производственной практики, прохождении государственной итоговой аттестации.

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

3.1. Распределение учебных часов по темам дисциплины и видам занятий

Наименование тем (содержательных модулей)	Количество часов (очная/ заочная форма)				
	Всего	В том числе			
		Лекции	Лабор.	Практ (Семин)	СРС
Тема 1. Системы счисления, форматы данных	20	2	12	-	6
Тема 2. Определения С, типы данных, функции ввода вывода	12	4	2	-	6
Тема 3. Алгоритмы с разветвлениями и циклами	22	6	10	-	6
Тема 4. Алгоритмы обработки одномерных массивов	23	6	8	-	9
Тема 5. Алгоритмы обработки двумерных массивов	20	4	8	-	8
Тема 6. Указатели, функции, параметры функции	18	4	6	-	8
Тема 7. Работа с файлами	15	4	5	-	6
Тема 8. Обработка строк	16	4	6	-	6
Тема 9. Структуры, линейные списки	20	10	8	-	2
Тема 10. Указатели - динамические массивы	24	8	14	-	2
Тема 11. Указатели на функции	3	2	-	-	1
Тема 12. Рекурсивные функции	5	4	-	-	1
Тема 13. Многомодульные программы	30	6	23	-	1
Тема 14. Препроцессор	5	4	-	-	1
Контактная работа (дополнительная)	10				
Курсовая работа	27				27
Итого по видам занятий	270	68	102	-	90
Контроль	90				
ИТОГО:	**360**	**68**	**102**	**-**	**90**

Формирование компетенций в результате освоения тем дисциплины

Компетенции	Темы дисциплины, нацеленные на выработку компетенции
ОПК-1	Темы 1 - 10
ОПК-8	Темы 3 - 12
ОПК-9	Темы 5 - 14

3.2. Лекции

Тема 1. Системы счисления, форматы данных.

Содержание темы 1:

Структурная схема ЭВМ. Основные функциональные устройства ЭВМ и их характеристики. Принцип программного управления ЭВМ. Системы счисления, используемые в ЭВМ, переводы чисел. Форматы целых и вещественных чисел.

Литература к теме 1: [1-8]

Тема 2. Определения С, типы данных, функции ввода вывода.

Содержание темы 2:

Основные соглашения языка С: константы, комментарии, переменные, операции и их приоритет, основные операторы, типы данных и приведение типов, функции консольного ввода вывода.

Литература к теме 2: [1-8]

Тема 3. Алгоритмы с разветвлениями и циклами.

Содержание темы 3:

Свойства алгоритма, блок-схемный способ описания алгоритма, линейные алгоритмы. Алгоритмы с ветвлениями if-else. Циклы while do-while, обработка чисел, выделение цифр.

Литература к теме 3: [1-8]

Тема 4. Алгоритмы обработки одномерных массивов.

Содержание темы 4:

Массивы, обработка цифр в массиве. Алгоритмы обработки массивов: поиск min и max, обмен значений, определение среднего арифметического, поиск первого - последнего, сортировка, реверс, массив счетчиков.

Литература к теме 4: [1-8]

Тема 5. Алгоритмы обработки двумерных массивов.

Содержание темы 5:

Матрицы: поиск max-min, сортировка в строке и строк, сдвиги в строке, перенос элементов в начало-конец строки или столбца.

Литература к теме 5: [1-8]

Тема 6. Указатели, функции, параметры функции.

Содержание темы 6:

Указатели, функции, параметры, массив – параметр функции, удаление из массива, вставка в массив, перестановки элементов в массиве. Указатели – параметры функций.

Литература к теме 6: [1-8]

Тема 7. Работа с файлами.

Содержание темы 7:

Текстовые и бинарные файлы, их особенности. Функции и способы обработки файлов

Литература к теме 7: [1-8]

Тема 8. Работа со строками.

Содержание темы 8:

Строки – массив символов. Алгоритмы обработки строк. Функции для работы со строками.

Литература к теме 8: [1-8]

Тема 9. Структуры, линейные списки.

Содержание темы 9:

Структуры, списки. Стек, очередь, ДЕК. Построение, дисциплины обслуживания, обработка.

Литература к теме 9: [1-8]

Тема 10. Указатели - динамические массивы.

Содержание темы 10:

Динамические переменные и массивы. Указатели на динамические массивы. Статическая, динамическая матрицы, массив указателей. Указатель на указатель, многомерные массивы.

Литература к теме 10: [1-8]

Тема 11. Указатели на функции.

Содержание темы 11:

Указатели на функции. Массивы указателей на функции. Указатели на функции, как параметры функций.

Литература к теме 11: [1-8]

Тема 12. Рекурсивные функции.

Содержание темы 12:

Принципы написания рекурсивных функций. Типы рекурсий. Примеры использования рекурсивных функций.

Литература к теме 12: [1-8]

Тема 13. Многомодульные программы.

Содержание темы 13:

Область действия имен. Многомодульные программы. Заголовочные файлы. Статическая библиотека.

Литература к теме 13: [1-8]

Тема 14. Препроцессор.

Содержание темы 14:

Назначение препроцессора. Назначение и использование основных директив препроцессора при написании программ.

Литература к теме 14: [1-8]

3.3. Практические (семинарские) занятия

В учебном плане не запланировано.

3.4. Лабораторные работы

№ п/п	Тема работы	Объем, час. очн/заочн	Литература
1	Перевод чисел из 10сс в 2, 8 , 16	2	[8,10]
2	Перевод чисел из 2сс в 8 и 16	2	[8,10]
3	Форматы целых чисел.	2	[8,10]
4	Форматы вещественных чисел	2	[8,10]
5	Сложение и вычитание целых чисел в дополнительном коде	2	[8,10]
6	Логические операции	2	[8,10]
7	Форматы ввода-печати	2	[8,11]
8	Сортировка(a,в,c,d)	2	[8,11]
9	Программы с разветвлениями, геометрические задачи	4	[8,11]
10	Выделение цифр из числа, обработка без массивов	4	[8,11]
11	Обработка одномерных массивов	8	[8,11]
12	Обработка двумерных массивов	8	[8,11]
13	Составление функций для обработки массивов	6	[8,11]
14	Обработка текстовых файлов	5	[8,11]
15	Обработка строк	6	[8,12]
16	Обработка линейных списков	8	[8,12]
17	Обработка динамических массивов	6	[8,12]
18	Обработка динамических матриц	8	[8,12]
19	Многомодульные программы формирования линейных списков	6	[8,9,12]
20	Многомодульные программы для обработки линейных списков	10	[8,9,12]
21	Многомодульные программы для формирования отчетов по данным из линейных списков	7	[8,9,12]
Итого:		**102**	

3.5. Самостоятельная работа студента

№ п/п	Виды самостоятельной работы студента	Объем, час. очн/ заочн
1	Изучение лекционного материала	33
2	Подготовка к практическим занятиям	-
3	Подготовка к лабораторным работам	30
4	Выполнение курсового проекта	-
5	Выполнение курсовой работы	27
6	Выполнение индивидуального задания	-
Итого:		90

3.6. Курсовой проект (работа), индивидуальное задание

Тематика курсовой работы связана с созданием программы административно-учетной системы (АУС), называемых также автоматизированными системами управления (АСУ), объектом машинной обработки является документ (группа документов). В составе задания по курсовой работе студент получает список параметров документов, полей хранимой информации и требуемых печатных документов. После составления тестового примера данных АУС студент разрабатывает структуру записи архивного файла и кодификатора, составляет программы обработки данных (ввод, редактирование, удаление, сортировка, формирование отчетов) на языке С и проверяет работу АУС на тестовых примерах.

В тестовых примерах должно быть проверено корректность выполнения отдельных режимов работы АУС.

ПО АУС разрабатывается как многомодульная программа, ее интерфейс выполняется, как правило, в текстовом режиме работы экрана.

Цель выполнения курсовой работы: освоение студентами полноэтапного процесса разработки многомодульной программы среднего уровня сложности - от изучения постановки задачи до получения и защиты разработанного проекта.

В результате выполнения курсовой работы студенты должны:

- Приобрести умение разрабатывать структуру программного проекта с выделением последовательных этапов его выполнения;
- Приобрести навыки разработки и тестирования многомодульной программы среднего уровня сложности;
- Освоить начальную методику разработки интерфейса пользователя в текстовом или графическом режиме;
- Освоить требования действующих стандартов на научно-технические отчеты.

4 ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

4.1 Критерии и шкалы для интегрированной оценки уровня сформированности компетенций

Составляющая компетенции – полнота знаний

- нулевой уровень: неверные, не аргументированные, с множеством грубых ошибок ответы на вопросы / ответы на три вопроса из четырех полностью отсутствуют. Уровень знаний ниже минимальных требований;
- минимальный уровень: даны не полные, не точные и аргументированные ответы на вопросы. Уровень знаний ниже минимальных требований. Допущено много грубых ошибок;
- пороговый уровень: даны недостаточно полные, точные и аргументированные ответы на вопросы. Плохо знает термины, определения и понятия; основные закономерности, соотношения, принципы. Допущено много негрубых ошибок;
- средний уровень: Даны достаточно полные, точные и аргументированные ответы на вопросы. В целом знает термины, определения и понятия; основные закономерности, соотношения, принципы. Допущено несколько негрубых ошибок;
- продвинутый уровень: даны полные, точные и аргументированные ответы на вопросы. Знает термины, определения и понятия; основные закономерности, соотношения, принципы. Допущено несколько негрубых ошибок;
- высокий уровень: даны полные, точные и аргументированные ответы на вопросы. Знает термины, определения и понятия; основные закономерности, соотношения, принципы. Допущено несколько неточностей.

Составляющая компетенции – умения

- нулевой уровень: полное отсутствие понимания сути методики решения задачи, допущено множество грубейших ошибок / задания не выполнены вообще;
- минимальный уровень: слабое понимание сути методики решения задачи, допущены грубые ошибки. Решения не обоснованы. Не умеет использовать нормативно-техническую литературу. Не ориентируется в специальной научной литературе, нормативно-правовых актах;
- пороговый уровень: достаточное понимание сути методики решения задачи, допущены ошибки. Решения не всегда обоснованы. Умеет использовать нормативно-техническую литературу. Слабо ориентируется в специальной научной литературе, нормативно-правовых актах;
- средний уровень: в целом понимает суть методики решения задачи, допущены ошибки. Решения не всегда обоснованы. Умеет использовать нормативно-техническую и специальную научную литературу, нормативно-правовые акты;
- продвинутый уровень: в целом понимает суть методики решения задачи, допущены неточности. Способен обосновать решения. Умеет использовать

нормативно-техническую и специальную научную литературу, нормативно-правовые акты;
- высокий уровень: Понимает суть методики решения задачи. Способен обосновать решения. Умеет использовать нормативно-техническую и специальную научную литературу, передовой зарубежный опыт, нормативно-правовые акты.

Составляющая компетенции – владение навыками

- нулевой уровень: не продемонстрировал навыки выполнения профессиональных задач. Испытывает существенные трудности при выполнении отдельных заданий;
- минимальный уровень: не продемонстрировал навыки выполнения профессиональных задач. Испытывает существенные трудности при выполнении отдельных заданий;
- пороговый уровень: владеет опытом готовности к профессиональной деятельности и профессиональному самосовершенствованию на пороговом уровне. Трудовые действия выполняет медленно и некачественно;
- средний уровень: владеет средним опытом готовности к профессиональной деятельности и профессиональному самосовершенствованию. Трудовые действия выполняет на среднем уровне по быстроте и качеству;
- продвинутый уровень: владеет опытом и достаточно выраженной личностной готовности к профессиональной деятельности и профессиональному самосовершенствованию. Быстро и качественно выполняет трудовые действия;
- высокий уровень: владеет опытом и выраженностью личностной готовности к профессиональной деятельности и профессиональному самосовершенствованию. Быстро и качественно выполняет трудовые действия.

Обобщенная оценка сформированности компетенций

- нулевой уровень: компетенции не сформированы;
- минимальный уровень: значительное количество компетенций не сформировано;
- пороговый уровень: все компетенции сформированы, но большинство на пороговом уровне;
- средний уровень: все компетенции сформированы на среднем уровне;
- продвинутый уровень: все компетенции сформированы на среднем или высоком уровне;
- высокий уровень: все компетенции сформированы на высоком уровне.

4.2 Вопросы к экзамену и пример экзаменационного билета (семестр 1)

1.Десятичное число, включающее целую и дробную части, перевести в 2с.с. с заданной точностью и выполнить проверку путем обратного перевода. Уметь пояснить термины: основание системы счисления, весовые коэффициенты, показатели степени при весовых коэффициентах. Уметь

объяснить допустимую величину абсолютной погрешности перевода дробной части числа. Например, число 28.43 перевести в 2с.с. с точностью 0.05. Выполнить проверку.

2. Число 128.34 перевести в 8с.с. с точностью 0.02. Выполнить проверку.

3. Число 88.14 перевести в 16с.с. с точностью 0.04. Выполнить проверку.

4.Системы счисления 2с.с., 8с.с., 16с.с., их назначение и использование в информатике.

5. Переводы чисел из 2с.с. в 8с.с. и обратно. Переводы чисел из 2с.с. в 16с.с. и обратно. Пример 1. Выполнить перевод десятичного числа 45.73 с точностью 0.02 по схеме: 10с.с. – 8с.с. – 16с.с – 10с.с.. и показать правильность перевода. Пример 2. Выполнить перевод десятичного числа 51.37 с точностью 0.04 по схеме: 10с.с. – 16с.с. – 8с.с – 10с.с.. и показать правильность перевода.

6.Правила записи и диапазон значений вещественных констант на языке Си. Привести примеры записи вещественных констант в форме с фиксированной точкой и в форме с экспонентой.

7.Представление символьных констант в языке Си. Кодировка символов в таблице 1251. Коды управляющих символов. Генерация кодов управляющих символов. Кодировка символов, имеющих изображение в левой половине таблицы. Символы правой половины таблицы 1251.

8. Правила записи и диапазон значений целых констант на языке Си. Привести примеры целых констант в 10с.с., 8с.с., 16с .с...

9.Форма записи символов в Esc- последовательностях в 8с.с. и в 16с.с.. Примеры записи символов.

10. Целочисленные типы языка Си. Диапазоны значений целых типов, назначение целых типов. Привести численные примеры. Пример 1. Числа 43 и -204 записать в форматах signed char и long соответственно в 2с.с., 8с.с. и 16с.с. (Первое из чисел записывается в формате signed char, а второе число записывается в формате long).

11. Сложение и вычитание целых чисел в 2с.с. в формате signed char. Пример задания. Два числа A=-43 и B= 68 записать формате signed char в 2с.с.. Выполнить операции C=A+B и E= B-A и проверить результаты.

12.Строковые константы. Представление строк в памяти. Правила конкатенации (склейки) строковых констант в программе на языке Си.

13.Вещественные типы данных в языке Си. Размеры, диапазоны и точность задания вещественных чисел в форматах float, double, long double.

14. Представление вещественного числа в формате float. Пример. Десятичное число -24.25 записать в формате float в 2с.с. и 16с.с..

15.Главное правило вычисления выражения при использовании операндов различных типов. Явное наложение типа, операция (тип). Примеры выражений.

16.Определение алгоритма. Основные свойства алгоритма. Формы записи алгоритмов.

17.Алгоритмы на обработку чисел без массивов:

1)Дано целое десятичное число a. Определить количество четных цифр в числе a.

2) Дано целое десятичное число а. Определить количество цифр в числе а, которые больше или равны числу 5.

3)Дано целое десятичное число а. Определить сумму цифр в числе а.

4) Дано целое десятичное число а. Определить сумму нечетных цифр в числе а.

5)Дано целое десятичное число а. Удалить из представления числа а цифры 2 и 5.

6) Дано целое десятичное число а. Заменить в представления числа а цифры 6 и 7 нулями.

7) Дано целое десятичное число а. Определить произведение цифр в числе а.

8)Дано целое десятичное число а (а>=2). Проверить, делиться ли число а на какое либо из чисел в диапазоне [2..а/2].

9) Дано целое десятичное число а. Проверить является число а кратным и 4 и 5.

10)Даны целые десятичные числа а и b. Если сумма цифр числа а больше суммы цифр числа b, обменять а и b местами.

18.Массивы как структурированные переменные. Примеры описания массивов.

19. Алгоритмы на обработку чисел с использованием массивов:

1)Дано целое десятичное число А. Обменять местами первую и последнюю цифры числа.

2)Дано целое десятичное число А. Обменять местами две младшие цифры числа.

3)Дано целое десятичное число А. Обменять местами две старшие цифры числа.

4)Дано целое десятичное число А. Сформировать массив десятичных цифр числа. В массиве цифр выполнить вычитание 1. Из массива цифр сформировать преобразованное число А.

5)Дано целое десятичное число А. Сформировать массив десятичных цифр числа. В массиве цифр выполнить сложение с 1. Из массива цифр сформировать преобразованное число А.

6)Дано целое десятичное число А. Сформировать массив двоичных цифр числа. В массиве цифр выполнить вычитание 1. Из массива цифр сформировать преобразованное число А.

7)Дано целое десятичное число А. Сформировать массив двоичных цифр числа. В массиве цифр выполнить сложение с 1. Из массива цифр сформировать преобразованное число А.

8)Дано целое десятичное число А. Сформировать массив восьмеричных цифр числа. В массиве цифр выполнить вычитание 1. Из массива цифр сформировать преобразованное число А.

9)Дано целое десятичное число А. Сформировать массив восьмеричных цифр числа. В массиве цифр выполнить сложение с 1. Из массива цифр сформировать преобразованное число А.

20.Инициализация элементов массива в операторе описания. Примеры описания массивов с инициализацией элементов.

21.Правило инициализации по умолчанию локальных и глобальных переменных программы.

22.Алгоритмы на обработку числовых одномерных массивов.

1)Определить количество положительных элементов массива.

2)Определить максимальный элемент и его позицию.

3)Обменять максимальный и минимальный элементы.

4)Определить позицию и значение последнего отрицательного элемента.

5)Определить среднее арифметическое отрицательных элементов.

6) Определить среднее геометрическое положительных элементов.

7)Выполнить сортировку положительных элементов массива. Позиции элементов массива, значения которых меньше или равно 0, не меняются.

8)Элементы массива расположить в следующем порядке: с начала отрицательные по возрастанию, затем положительные по убыванию, нулевые в конце массива.

9)Определить минимальный элемент массива среди положительных элементов.

10)Обменять местами первый и последний четные элементы. Массив целочисленный.

11)Определить цифры, которые не встречаются в представлении целого числа А.

12)Определить цифры, которые встречаются в представлении целого числа А один раз.

13)Определить цифру, которая встречаются в представлении целого числа А максимальное количество раз.

14) Элементы массива расположить в следующем порядке: с начала положительные по убыванию, затем отрицательные по возрастанию, нулевые в конце массива.

23.Определение двумерных массивов (матриц). Описание матриц с инициализацией элементов. Расположение в памяти элементов матрицы.

24.Алгоритмы на обработку матриц:

1)В прямоугольной матрице определить номера строк, у которых значение среднего арифметического элементов отрицательная величина. Для каждой такой строки вывести на экран номер строки и значение среднего арифметического.

2)В каждом столбце прямоугольной матрицы нулевые элементы перенести в конец строки. Сохранить исходный порядок следования ненулевых элементов.

3)В каждом столбце прямоугольной матрицы расположить элементы в обратном порядке. При этом положение нулевых элементов не менять.

4)Элементы строк с четными номерами расположить в порядке возрастания. Строки матрицы пронумерованы с нуля.

5)В каждом столбце прямоугольной матрицы проверить значение первого элемента. Если первый элемент отрицательный, то элементы столбца расположить по убыванию.

6)В каждом столбце прямоугольной матрицы элементы сдвинуть циклически вниз на две позиции.

7)В каждом строке прямоугольной матрицы элементы сдвинуть циклически вправо на две позиции.

8)В каждой строке прямоугольной матрицы определить количество положительных элементов. Если таких элементов в строке больше половины, то выводить на экран номер строки и количество положительных элементов.

9)В прямоугольной матрице определить количество столбцов, состоящих из отрицательных элементов.

10)В прямоугольной матрице определить номера столбцов (вывести на экран), в которых нет нулевых элементов.

25.Определение и описание типизированных (нетипизированных) указателей. Операция * для описания указателя и разыменовывания указателя. Операция & взятия адреса. Вывод на экран значения указателя и вывод значения полей, которые адресуются указателями.

26.Инициализация переменных-указателей. Константа NULL.

27.Массивы и указатели. Использование указателей для доступа к элементам массивов.

28.Организация функций языка Си. Структура функции. Параметры функций (параметры-указатели и параметры-значения, входные и выходные параметры). Понятия «формальные» и «фактические» параметры функции. Тип функции и оператор return <имя переменной>. Операция «()» для описания функции и обращения к функции. Примеры программ с использованием функций.

29.Спецификатор const для описания указателей. Использование спецификатора const для описания параметров-указателей функции. Примеры функций.

30.Прототипы функций их назначение и описание в программе Си.

31.Определение файла. Текстовые и бинарные файлы. Стандартные потоки

32.Сравнительный анализ текстовых и бинарных файлов.

33.Функция fopen() (fopen_s()) для создания и открытия файлов. Описание и назначение файловой переменной. Описание и назначение функции fopen() (fopen_s()). Константа EOF.

34.Форматный ввод-вывод в текстовый файл, функции fscanf(), fprintf(). Чтение чисел из текстового файла и запись чисел в текстовый файл.

35.Посимвольное чтение и запись текстовых файлов. Примеры программ: запись текстового файла с клавиатуры, вывод текстового файла на экран, копирования файла.

36.Функции fread(), fwrite() для чтения/записи бинарных файлов.

37.Функции для позиционирования указателя файла: fseek(), ftell(), rewind().

Пример экзаменационного билета

ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Программа подготовки: бакалавриат
Направление подготовки: 09.03.01 «Информатика и вычислительная техника»
Профиль: Системное программирование. Семестр: 1
Учебная дисциплина: Программирование

Билет N 1 (суммарная оценка 100 баллов)

1.Разработать схему алгоритма и программу на языке Си. Выполнить ввод исходных данных с консоли. Результаты программы вывести на консоль. В отчете представить схему алгоритма программы, программу на языке Си, тестовый пример исходных данных и результатов программы. **Алгоритм обработки**: дано целое десятичное число А. Сформировать массив десятичных цифр числа А. В массиве цифр выполнить добавление 1 к четным цифрам, вычитание 1 из нечетных цифр. Вывести массив цифр до и после обмена. Из массива цифр сформировать десятичное число. (50 баллов).

2. Диагностика ошибок с использованием функции perror(). (10 баллов).

3. Инициализация переменных-указателей. Константа NULL. (20 баллов).

4. Десятичное число -24.25 записать в формате float в 2с.с. и 16с.с.(20 баллов).

КРИТЕРИИ
оценивания экзаменационной работы

по дисциплине «Программирование» для обучающихся по направлению подготовки: 09.03.01 «Информатика и вычислительная техника» (профиль – Системное программирование.)

Экзамен проводится письменно по билетам. Билет содержит 4 вопроса, включающий теоретические положения и практические навыки в виде составления алгоритма и программ. При необходимости отвечающий должен сопроводить выбранный алгоритм поясняющей блок-схемой (рисунком).

Вопросы охватывают теоретическую часть курса, а также требуют демонстрации практических навыков составления программ, полученных студентом в ходе лабораторных работ.

Правильные ответы на вопросы оценивается в: 1-й вопрос 50 баллов, 2-й вопрос 10 баллов, 3-й вопрос 20 баллов, 4-й вопрос 20 баллов. Если ответ не снабжен достаточной иллюстративностью снимается 5-20 баллов. За ошибки в формулах, алгоритмах обработки, описании функций снимается до 20 баллов. При отсутствии правильного ответа на поставленный вопрос обучающийся получает ноль баллов. Полученные баллы за ответы на вопросы билета суммируются, и выводится итоговая оценка по 100-балльной шкале.

Полученная оценка по 100-балльной шкале определяет оценку по

государственной шкале и шкале ECTS:

Сумма баллов по 100-бальной шкале	Оценка по шкале ECTS	Оценка по государственной шкале
90-100	A	Отлично
80-89	B	Хорошо
75-79	C	
70-74	D	Удовлетворительно
60-69	E	
35-59	FX	Неудовлетворительно
0-34	F*	

* – с обязательным повторным изучением дисциплины.

4.3 Вопросы к экзамену и пример экзаменационного билета (семестр 2)

Часть 1

1.Функции malloc() и free() для работы с динамической памятью. Пример выделения памяти динамической переменной и освобождения памяти.

2.Определение и описание структур. Пример описание структуры. Обращение к полям структуры с использованием уточненного имени.

3.Определение и описание структуры, использующей вложенные структуры. Пример описание структуры.

4.Использование указателей для работы со структурами. Операция «->» - обращение к элементу структуры с использованием указателя. Пример использования.

5.Глобальные переменные, порядок определения. Инициализации глобальных переменных в операторе определения, инициализация по умолчанию.

6.Определение и описание массива структур. Пример описания.

7.Область видимости глобальных переменных. Различия в понятиях определение глобальной переменной и объявление глобальной переменной. Примеры экспорта глобальной переменной из одного файла программы в другой файл.

8.Функции языка Си как глобальные и внешние объекты программы.

9.Примеры использования функций в нескольких файлах программы. Применение спецификатора static для определения функций и глобальных переменных программы.

10.Автоматические переменные и параметры функции.

11. Регистровые переменные и параметры функции.

12.Статические локальные переменные функции.

13.Продолжительность жизни функций и переменных программы: статическая, локальная, динамическая.

14.Обработка статических и динамических одномерных массивов. Создание и удаление динамического массива.

15. Обработка статических и динамических матриц. Создание и удаление динамических матриц.

16.Определение переменной указателя на матрицу.

17.Описание переменных указателей для работы с многомерными массивами.

Часть 2

1.Массивы указателей. Использование массива указателей для построения динамической матрицы с переменным количеством элементов в строках. Привести пример.

2.Массивы указателей. Использование массива указателей для построения прямоугольной динамической матрицы (отметить ограничения на количество строк и столбцов). Привести пример.

3.Алгоритм освобождения динамической памяти, которая используется прямоугольной динамической матрицей, построенной на базе массива указателей

4.Организация свободного массива строк. Привести пример.

5.Указатель на указатель. Использование массива указателей для построения прямоугольной динамической матрицы (отметить ограничения на количество строк и столбцов). Привести пример.

6.Алгоритм освобождения динамической памяти, которая используется прямоугольной динамической матрицей, построенной на базе указателя на указатель.

7.Особенности алгоритма сортировки строк прямоугольной динамической матрицы, построенной на базе переменной указатель на указатель.

8.Моделирование многомерных массивов на базе переменной указатель на указатель.

9.Указатель на функцию. Способы вызова функции с помощью указателя на функцию.

10.Оператор мультиветвления switch. Привести пример.

11.Массивы указателей на функции. Использование массива указателей для мультиветвления в программе.

12.Указатель на функцию, как параметр функции. Привести пример.

13.Статическая библиотека, назначение и порядок использования.

14.Пример построения статической библиотеки.

15.Препроцессор языка Си. Назначение и порядок использования.

16.Назначение и состав заголовочных файлов языка Си. Подключение заголовочных файлов директивой #include, порядок использования угловых скобок и двойных кавычек для заголовочных файлов.

17.Использование директивы #define для описания макросов и строк подстановки. Последовательные макро подстановки в языке СИ. Привести пример.

18.Макросы с формальными параметрами. Привести пример описания и вызова макроса.

19.Достоинства и недостатки макросов с параметрами по отношению к функциям языка Си.

20.Директивы условной компиляции. Перечислить основные директивы. Привести пример.

21.Рекурсивные функции. Прямая и косвенная рекурсия.

22.Действия компилятора при выполнении рекурсивных вызовов. Привести пример.

23. Когда и зачем применяют рекурсивные функции.

Примеры заданий по обработке строк.

Общее задание. В исходной строке записаны слова через один или несколько пробелов. В начале и в конце строки также могут быть пробелы. Переписать в результирующую строку слова через один пробел. В начале и конце строки пробелов быть не должно. Слово это последовательность символов не равных пробелу.

1. Переписать слова, длина которых не более 4 символов.

2. Переписать слова, длина которых больше 3 символов и не более 6 символов.

3. Переписать слова, которые начинаются символом ‘a’.

4. Переписать слова, которые заканчиваются символом ‘z’.

5. Переписать слова длиной 5 символов, которые начинаются цифрой.

6. Переписать слова, которые представляют собой правильную целую десятичную константу.

7. Переписать слова, которые представляют собой правильную целую восьмеричную константу.

8. Переписать слова, которые представляют собой правильную целую шестнадцатеричную константу.

9. Переписать слово максимальной длины и слово минимальной длины.

10.Переписать все слова в порядке возрастания их длин.

11.Переписать все слова в порядке убывания их длин.

12.Переписать все слова в исходном порядке, при этом в каждом слове первый и последний символы обменять местами.

13.Переписать все слова, в которых имеются повторяющиеся символы. При отсутствии таковых результирующая строка пустая.

14.Переписать все слова, у которых первый и второй символы одинаковые. При отсутствии таковых результирующая строка пустая.

15.Переписать все слова палиндромы. Это слова, которые читаются слева направо и справа налево одинаково. При отсутствии таковых результирующая строка пустая.

Примеры заданий по спискам

1.Если в стеке находится больше чем один элемент, то выполнить попарный обмен элементов этого стека (первый элемент обменять местами со вторым элементом, третий - с четвертым и т.д.).

2.Определить, упорядочены ли элементы очереди по возрастанию и, если это так, то вставить в начало очереди элемент, который равняется полусумме первого и последнего элементов.

3.Определить количество элементов очереди, которые имеют заданное значение k и, если такие элементы существуют, то изъять последний из них. Первый и последний элементы очереди не могут быть равными k.

4.Определить количество элементов очереди, значения которых совпадают со значением последнего элемента и, если такие элементы существуют, то удалить первый из них. Первый и последний элементы очереди не могут быть равными.

5.По заданному стеку создать новый стек, перенеся в него положительные элементы в обратном порядке.

6.Добавить новый элемент в очередь: если значение нового элемента четное, то установить его в конце очереди, в противном случае - в начале очереди. Значение нового элемента ввести из клавиатуры.

7.Вставить в очередь два новых элемента: один из них - перед последним элементом исходной очереди, второй - после последнего элемента. Значения новых элементов установить равными удвоенному значению первого элемента очереди.

8.Если в очереди имеются элементы с заданным значением q, то обменять первый такой элемент с последним элементом данной очереди.

9.Удалить из стека последний отрицательный элемент. Первый и последний элементы стека – положительные.

10.В очереди первый и последний элементы – нулевые. Удалить из очереди первый положительный элемент.

11.В начало очереди вставить введенное с клавиатуры q, а перед ним число k, которое равняется количеству элементов очереди, равных q.

12.В очереди имеется единственный нулевой элемент, который разделяет ее на две части. Сформировать две новые очереди, записав в них соответственно первую и вторую части исходной очереди. Нулевой элемент не может быть ни первым, ни последним в исходной очереди.

Пример экзаменационного билета

ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Программа подготовки: бакалавриат
Направление подготовки: 09.03.01 «Информатика и вычислительная техника»
Профиль: Системное программирование. Семестр: 2
Учебная дисциплина: Программирование

Билет N 1 (суммарная оценка 100 баллов)

1. В программе определена переменная **double (*d)[6]; d=(double (*)[6])malloc(5*6*8);** Определить смещение в байтах при выполнении ***(d+2)+5.** Пояснить выполнение оператора ***(d[3]+5)=111.11;**(10 баллов)

2. Разработать программу по обработке строки с использованием линейных списков. Исходная строка вводится с клавиатуры. В строке записаны слова через один или несколько пробелов. В начале и в конце строки также могут быть пробелы. Выделить из строки слова и записать в линейный список. Обработку проводить в списке. Результат обработки выводится на экран (в консольное окно). Представить текст программы, пример исходной строки, схему списка, которая поясняет алгоритм выполнения задания, результат обработки на экране.

Алгоритм обработки строки: В очередь записать слова, начинающиеся с заглавной латинской буквы. Не изменяя положения первой буквы переставить буквы в обратном порядке, в словах, расположенных между 1-м и 3-м словами, начинающихся с буквы «Z» (50 баллов)

3. Разработать программу по обработке прямоугольной динамической матрицы, которая построена на базе указателя на указатель. В программе использовать следующие функции: создание динамической матрицы, ввод элементов матрицы с клавиатуры, вывод матрицы на экран, удаление динамической матрицы, а также функции обработки матрицы согласно следующему заданию.

В строках вещественной матрицы между первым и пятым отрицательными элементами расположить отрицательные элементы, затем нули, в конце положительные с сохранением исходного порядка. (40 баллов)

КРИТЕРИИ
оценивания экзаменационной работы

по дисциплине «Программирование» для обучающихся по направлению подготовки: 09.03.01 «Информатика и вычислительная техника» (профиль – Системное программирование.)

Экзамен проводится письменно по билетам. Билет содержит 3 вопроса, включающий теоретические положения и практические навыки в виде составления алгоритма и программ с использованием функций. При необходимости отвечающий должен сопроводить выбранный алгоритм поясняющей блок-схемой (рисунком).

Вопросы охватывают теоретическую часть курса, а также требуют демонстрации практических навыков, полученных студентом в ходе лабораторных работ.

Правильные ответы на вопросы оценивается в: 1-й вопрос 10 баллов, 2-й вопрос 50 баллов, 3-й вопрос 40 баллов. Если ответ не снабжен достаточной иллюстративностью снимается 5-20 баллов. За ошибки в формулах, алгоритмах обработки снимается до 20 баллов. При отсутствии правильного ответа на поставленный вопрос обучающийся получает ноль баллов. Полученные баллы за ответы на вопросы билета суммируются, и выводится итоговая оценка по 100-балльной шкале.

Полученная оценка по 100-балльной шкале определяет оценку по государственной шкале и шкале ECTS.

4.4 Пример текущего опроса на практических (семинарских) занятиях и лабораторных работах

На примере темы «Обработка линейных массивов»:

1. Алгоритм вычисления среднего арифметического массива
2. Использование переменных сумматоров и счетчиков
3. Отличие в алгоритмах поиска максимума в массиве и максимума среди отрицательных чисел массива?
4. Отличия поиска первого или последнего значения по заданному условию
5. Приведите блок-схему алгоритма реверса массива
6. В каком случае удобно применять массив счетчиков?
7. Сдвиги, удаление, вставки в массиве

4.5 Курсовое проектирование

Курсовая работа предусматривает разработку многомодульной программы, которая должна обеспечивать 9 режимов обработки информации. Каждый правильно написанный и отлаженный модуль оценивается в 8 баллов, итого разработка программных модулей максимально оценивается в 72 балла. За ошибки в программировании, неправильную обработку данных снимаются баллы.

При описании программного продукта в пояснительной записке требуется привести 5 блок-схем алгоритмов разрабатываемых модулей. Каждая правильно составленная блок-схема оценивается в 4 балла, максимальное количество баллов за разработку блок-схем составляет 20.

Оформление пояснительной записки в соответствии со стандартами и требованиями к оформлению документации на программное обеспечение оценивается в 8 баллов.

Общее количество баллов за выполнение курсовой работы – 100. На защите курсовой работы комиссия может снять дополнительные баллы, если студент не дает, или дает неправильные ответы на поставленные вопросы.

Полученная оценка по 100-балльной шкале определяет оценку по государственной шкале и шкале ECTS.

5 РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

I. Основная литература

1. Журавлева М.Г. Основы программирования. Введение в язык Си. Ч.1 : учебное пособие по курсам «Программирование», «Основы алгоритмизации и программирования» / Журавлева М.Г., Алексеев В.А., Домашнев П.А.. — Липецк : Липецкий государственный технический университет, ЭБС АСВ, 2019. — 99 с. — ISBN 978-5-00175-001-7. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL:

https://www.iprbookshop.ru/101463.html (дата обращения: 03.10.2021). — Режим доступа: для авторизир. Пользователей

2. Керниган Б.В. Язык программирования С : учебник / Керниган Б.В., Ричи Д.М.. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 313 с. — ISBN 978-5-4497-0918-9. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: https://www.iprbookshop.ru/102075.html (дата обращения: 03.10.2021). — Режим доступа: для авторизир. Пользователей

3. Фридман А.Л. Язык программирования С++ : учебное пособие / Фридман А.Л.. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 217 с. — ISBN 978-5-4497-0920-2. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: https://www.iprbookshop.ru/102076.html (дата обращения: 03.10.2021). — Режим доступа: для авторизир. Пользователей

4. Страуструп Б. Язык программирования С++ для профессионалов : учебник / Страуструп Б.. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 670 с. — ISBN 978-5-4497-0922-6. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: https://www.iprbookshop.ru/102077.html (дата обращения: 03.10.2021). — Режим доступа: для авторизир. Пользователей

II. Дополнительная литература

5. Златопольский Д.М. Программирование: типовые задачи, алгоритмы, методы / Златопольский Д.М.. — Москва : Лаборатория знаний, 2020. — 224 с. — ISBN 978-5-00101-789-9. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: https://www.iprbookshop.ru/12264.html (дата обращения: 25.12.2021). — Режим доступа: для авторизир. Пользователей

6. Алексеев Ю.Е. Разработка функций и модулей пользователя на языке С в среде VS C++. Модуль 3 : учебное пособие / Алексеев Ю.Е., Куров А.В.. — Москва : Московский государственный технический университет имени Н.Э. Баумана, 2020. — 114 с. — ISBN 978-5-7038-5340-5. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: https://www.iprbookshop.ru/115634.html (дата обращения: 25.12.2021). — Режим доступа: для авторизир. Пользователей

7. Курчеева Г.И. Информационные технологии в цифровой экономике : учебное пособие / Курчеева Г.И., Томилов И.Н.. — Новосибирск : Новосибирский государственный технический университет, 2019. — 79 с. — ISBN 978-5-7782-4037-7. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: https://www.iprbookshop.ru/98789.html (дата обращения: 25.12.2021). — Режим доступа: для авторизир. Пользователей

6 УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Учебно-методические издания, разработанные в ДонНТУ:

8. Курс лекций по дисциплине «Программирование» [Электронный ресурс]: для обучающихся по направлению подготовки 09.03.01 «Информатика и вычислительная техника» всех форм обучения / ГОУВПО «ДОННТУ»; Каф. компьютерной инженерии; сост. Л.И. Дорожко. - Донецк : ДОННТУ, 2021. – Систем. требования: Acrobat Reader. – (доступ через личный кабинет студента).

9. Методические рекомендации к выполнению курсовой работы по дисциплине «Программирование» [Электронный ресурс]: для обучающихся по направлениям подготовки 09.03.01 «Информатика и вычислительная техника» всех форм обучения / ГОУВПО «ДОННТУ», Каф. компьютерной инженерии; сост. Л. И. Дорожко. — Донецк : ДОННТУ, 2021. – Систем. требования: Acrobat Reader. – (доступ через личный кабинет студента)

10. Методические рекомендации к выполнению лабораторных работ по дисциплине «Программирование» Часть 1 [Электронный ресурс]: для обучающихся по направлению подготовки 09.03.01 «Информатика и вычислительная техника» всех форм обучения / ГОУВПО «ДОННТУ»; Каф. компьютерной инженерии; сост. Л.И. Дорожко, Н. С. Максименко, К. Б. Юсупова. - Донецк : ДОННТУ, 2021. – Систем. требования: Acrobat Reader. – (доступ через личный кабинет студента).

11. Методические рекомендации к выполнению лабораторных работ по дисциплине «Программирование» Часть 2 [Электронный ресурс]: для обучающихся по направлению подготовки 09.03.01 «Информатика и вычислительная техника» всех форм обучения / ГОУВПО «ДОННТУ»; Каф. компьютерной инженерии; сост. Л.И. Дорожко, Н. С. Максименко. - Донецк : ДОННТУ, 2021. – Систем. требования: Acrobat Reader. – (доступ через личный кабинет студента).

12. Методические рекомендации к выполнению лабораторных работ по дисциплине «Программирование» Часть 3 [Электронный ресурс]: для обучающихся по направлению подготовки 09.03.01 «Информатика и вычислительная техника» всех форм обучения / ГОУВПО «ДОННТУ»; Каф. компьютерной инженерии; сост. Л.И. Дорожко, Н. С. Максименко. - Донецк : ДОННТУ, 2021. – Систем. требования: Acrobat Reader. – (доступ через личный кабинет студента).

13. Методические рекомендации к самостоятельной работе студентов по дисциплине «Программирование» : для обучающихся по направлению подготовки 09.03.01 «Информатика и вычислительная техника» всех форм обучения / ГОУВПО «ДОННТУ», Каф. компьютерной инженерии ; сост. Л. И. Дорожко. – Донецк : ДОННТУ, 2021. – Систем. требования: Acrobat Reader. – (доступ через личный кабинет студента).

Периодические издания и образовательные ресурсы:

1. Информатика и кибернетика (2015-2017).
2. Вестник Донецкого национального технического университета (2016-2017).
3. Системный анализ и информационные технологии в науках о природе и обществе (2011-2017).
4. Научные труды Донецкого национального технического университета. Серия «Проблемы моделирования и автоматизации проектирования» (2008-2013)
5. Научные труды Донецкого национального технического университета. Серия «Информатика, кибернетика и вычислительная техника» (2008-2021).

Электронно-информационные ресурсы

1. Библиотека ДонНТУ http://ed.donntu.org
2. Электронно-библиотечная система - http://www.iprbookshop.ru/

7 МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

1. **Лекционные занятия:**
   Учебная аудитория №2.233 учебный корпус 2, для проведения занятий лекционного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации; (мультимедийное оборудование: ноутбук, Операционная система Microsoft Windows XP Libreoffice 5.3.4.(2017), мультимедийный проектор, экран; специализированная мебель: доска аудиторная, парты, демонстрационные стенды и плакаты.)
2. **Лабораторные работы:**
   Учебная аудитория №4.003 учебный корпус 4, для выполнения лабораторных работ, проведения групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Используется специализированная мебель: доска аудиторная, столы аудиторные, стулья ученические и компьютерное оборудование с установленной ОС Windows 7 Ultimate 64-bit SP1, AMD A4-6300, 3.7 GHz, 4,00ГБ. LibreOffice 5.2.2 (лицензия GNULGPLv3+ и MPL2.0), Mozilla Firefox лицензия GNU GPL и GNU LGPL, Notepad++ лицензия GNU GPL 2

3. **Самостоятельная работа:**
   Помещения для самостоятельной работы с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду организации: читальные залы, учебные корпуса 2,3

(Компьютерная техника с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду (ЭИОС ДОННТУ) и электронно-библиотечную систему (ЭБС IPRbooks), а также возможностью индивидуального неограниченного доступа обучающихся в ЭБС и ЭИОС посредством Wi-Fi с персональных мобильных устройств. ОС- Microsoft Windows 7, OpenOffice 2.0.3 – общественная лицензия MPL 2.0/ Grub loader for ALT Linux - лицензия GNU LGPL v3/ Mozilla Firefox - лицензия MPL2.0, Moodle (Modular Object-Oriented Dynamic Learning Environment) - лицензия GNU GPL.обучающихся в ЭБС и ЭИОС посредством Wi-Fi с персональных мобильных устройств