

УТВЕРЖДАЮ
 Декан факультета

 (подпись) Матвеев П.В.
 ФИО
 «___» _____ 20__

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ ИНФОРМАТИКА: ОСНОВЫ ПРОГРАММИРОВАНИЯ

Направление/специальность подготовки	09.03.04 Программная инженерия
Специализация/профиль/программа подготовки	Разработка программно-информационных систем
Уровень высшего образования	Бакалавриат
Форма обучения	Очная
Факультет	О Естественнонаучный
Выпускающая кафедра	О7 Информационные системы и программная инженерия
Кафедра-разработчик рабочей программы	О7 Информационные системы и программная инженерия

КУРС	СЕМЕСТР	ОБЩАЯ ТРУДОЁМКОСТЬ (ЗАЧЕТНЫХ ЕДИНИЦ)	ЧАСЫ (по наличию видов занятий)									ВИД ПРОМЕЖУТОЧНОГО КОНТРОЛЯ
			ОБЩАЯ ТРУДОЁМКОСТЬ	АУДИТОРНЫЕ ЗАНЯТИЯ				САМОСТОЯТЕЛЬНАЯ РАБОТА				
				ВСЕГО	ЛЕКЦИИ	ЛАБОРАТОРНЫЙ ПРАКТИКУМ	ПРАКТИЧЕСКИЕ ЗАНЯТИЯ	ВСЕГО	КУРСОВОЙ ПРОЕКТ	КУРСОВАЯ РАБОТА	ДРУГИЕ ВИДЫ САМОСТ. РАБОТЫ	
1	1	5	180	68	34	0	34	112	0	0	112	ЭКЗ.

ЛИСТ СОГЛАСОВАНИЯ

РАБОЧАЯ ПРОГРАММА СОСТАВЛЕНА В СООТВЕТСТВИИ С ТРЕБОВАНИЯМИ ФЕДЕРАЛЬНОГО
ГОСУДАРСТВЕННОГО ОБРАЗОВАТЕЛЬНОГО СТАНДАРТА ВЫСШЕГО ОБРАЗОВАНИЯ (ФГОС ВО)

09.03.04 Программная инженерия

год набора группы: 2024

Программу составил:

Кафедра О7 Информационные системы и программная инженерия
Палехова Ольга Александровна, старший преподаватель

Программа рассмотрена
на заседании кафедры-разработчика
рабочей программы **О7 Информационные системы и программная инженерия**

Заведующий кафедрой Семенова Е.Г., д.т.н., проф.

Программа рассмотрена
на заседании выпускающей кафедры

О7 Информационные системы и программная инженерия

Заведующий кафедрой Семенова Е.Г., д.т.н., проф.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ ИНФОРМАТИКА: ОСНОВЫ ПРОГРАММИРОВАНИЯ

Разделы рабочей программы

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ
2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВО
3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ
4. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ
5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ
6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Приложения к рабочей программе дисциплины

- Приложение 1. Аннотация рабочей программы
- Приложение 2. Технологии и формы обучения
- Приложение 3. Фонды оценочных средств

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины является формирование следующих компетенций:

ОПК-2 — способность понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности

ОПК-6 — способность разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов

Формированию компетенций служит достижение следующих результатов образования:

ОПК-2

знания:

способы представления данных различных типов в памяти компьютера и принципы их обработки;

жизненный цикл программы;

базовые алгоритмические конструкции;

современные парадигмы программирования;

умения:

чтение и анализ текстов программ, написанных на языке программирования Си;

формирование отчетной документации с помощью текстовых процессоров;

построение схем алгоритмов с помощью сервиса draw.io;

навыки:

выбора типа скалярных данных с учетом ограничений, накладываемых способом их кодирования;

работы в интегрированных средах разработки программ;

анализа значений и адресов переменных в памяти при отладке программ.

ОПК-6

знания:

современных парадигм программирования;

понятия алгоритма, свойств алгоритма;

базовых алгоритмических конструкций;

способов представления данных различных типов в памяти компьютера и принципов их обработки;

типовых алгоритмов решения вычислительных задач;

типовых алгоритмов обработки массивов данных с произвольным и последовательным доступом к данным;

принципов нисходящего и восходящего проектирования;

синтаксиса и семантики языка Си;

умения:

разрабатывать алгоритмы решения вычислительных задач и задач обработки массивов данных с произвольным и последовательным доступом к данным;

применять принцип нисходящего проектирования к разработке алгоритмов;

представлять алгоритмы в графическом виде в соответствии с ГОСТ 19.701-90 и в виде текстов программ на языке Си;

определять вычислительную и пространственную сложность алгоритмов;

составлять тестовые наборы для ручного тестирования программ;

навыки:

выбора типа скалярных данных с учетом ограничений, накладываемых способом их кодирования;

анализа значений и адресов переменных в памяти при отладке программ;

выбора оптимального алгоритма обработки массивов данных для решения типовых задач;

написания текстов программ на языке программирования Си на основе типовых алгоритмов обработки данных произвольного и последовательного доступа.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВО

Дисциплина **ИНФОРМАТИКА: ОСНОВЫ ПРОГРАММИРОВАНИЯ** является дисциплиной **обязательной части блока 1** программы подготовки по направлению *09.03.04 Программная инженерия*.

Содержание дисциплины является логическим продолжением содержания школьных курсов и служит основой для освоения дисциплин: **ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ПРОГРАММИРОВАНИЕ, СТРУКТУРЫ И ОРГАНИЗАЦИЯ ДАННЫХ**

Требования к уровню подготовки обучающихся и предварительные компетенции определены Федеральным государственным образовательным стандартом среднего общего образования.

3. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины составляет 5 з.е., 180 ч.

3.1. Содержание (дидактика) дисциплины

КУРС	СЕМЕСТР	Наименование разделов и дидактических единиц	ВСЕГО	Аудиторные занятия в контактной форме			Самостоятельная работа студентов	Формируемая компетенция, %	
				ВСЕГО	Лекции	Практические занятия		ОПК-2	ОПК-6
1	1	Раздел 1. Введение. 1.1. Решение задач на компьютере. 1.2. Современные парадигмы программирования. 1.3. Языки программирования, классификация языков программирования. 1.4. Этапы процесса разработки программ. 1.5. Системы программирования, трансляторы. 1.6. Алгоритм, его свойства и способы записи.	6	4	2	2	2	5	5
1	1	Раздел 2. Основы языка Си. Программирование базовых алгоритмических конструкций на языке Си. 2.1. Основные компоненты языка: алфавит, слова, идентификаторы, выражения, операторы. 2.2. Лексемы, классы лексем. 2.3. Константы, переменные, выражения. Правила преобразования типов. 2.4. Операторы языка Си.Линейный алгоритм. 2.5. Разветвляющиеся алгоритмы. 2.6. Способы программирования ветвлений на языке Си. 2.7. Решение типовых задач с ветвлением. 2.8. Циклические алгоритмы. 2.9. Программирование циклов на языке Си. 2.10. Решение типовых задач с циклами.	34	14	8	6	20	20	20
1	1	Раздел 3. Типы данных языка Си: скалярные типы, указатели, массивы. Распределение памяти под данные. 3.1. Представление информации в компьютере и типы данных языка Си. 3.2. Понятие указателя. Объявление и инициализация указателей. Операции над указателями. 3.3. Особенности использования указателей. 3.4. Одномерные, двумерные и многомерные массивы. Объявление и способы инициализации. 3.5. Использование указателей при работе с массивами. 3.6. Динамическое выделение памяти. 3.7. Решение типовых задач с использованием массивов.	44	14	6	8	30	30	15
1	1	Раздел 4. Вспомогательные алгоритмы. Функции. 4.1. Деление задачи на подзадачи. Вспомогательные алгоритмы. Технология нисходящего и восходящего программирования. 4.2. Оценка сложности алгоритмов. 4.3. Определение и описание функций в языке Си. Прототип. Раздельная компиляция исходных файлов. Создание и использование статических библиотек. 4.4. Входные данные. Формальные и фактические параметры. Механизм передачи параметров. 4.5. Выходные данные. Оператор return. Использование побочного эффекта функции. 4.6. Указатель на функцию. 4.7. Время жизни и область видимости переменных. Классы памяти. 4.8. Рекурсивные функции. 4.9. Параметры функции main(). 4.10. Функции с переменным числом параметров. 4.11. Примеры решения задач.	64	24	12	12	40	20	40
1	1	Раздел 5. Структурирование данных. 5.1. Символьные массивы, Си-строка. Определение, объявление, инициализация, особенности ввода и вывода строк. Операции со строками, библиотечные функции для работы со строками. 5.2. Агрегирование данных разных типов. Структуры. Указатели на структуры. Операции со структурами. 5.3. Объединения. 5.4. Файл. Виды файлов. Потоки. Режимы открытия. Функции чтения и записи данных в файл. Особенности обработки текстовых и бинарных файлов.	32	12	6	6	20	25	20
Всего за 1 семестр			180	68	34	34	112	100	100
Всего по дисциплине			180	68	34	34	112	100	100

3.2. Аудиторный практикум

№ п/п	Номер и наименование раздела дисциплины	Тема практического занятия	Объем, ауд. часов
1	Раздел 1. Введение.	Системы программирования. Создание проекта.	2
2	Раздел 2. Основы языка Си. Программирование базовых алгоритмических конструкций на языке Си.	Программирование базовых алгоритмических структур	6
3	Раздел 3. Типы данных языка Си: скалярные типы, указатели, массивы. Распределение памяти под данные.	Указатели. Динамическое выделение памяти.	8
4	Раздел 4. Вспомогательные алгоритмы. Функции.	Вспомогательные алгоритмы. Функции. Создание статических библиотек	10
5		Оценка эффективности алгоритмов	2
6	Раздел 5. Структурирование данных.	Обработка файлов средствами языка Си	6
Всего за 1 семестр			34

3.3. Самостоятельная работа студента (СРС)

№ п/п	Номер и наименование раздела дисциплины	Содержание учебного задания	Объем, часов
1	Раздел 1. Введение.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	1
2		Выбор системы программирования и установка ее на домашнем компьютере	1
3	Раздел 2. Основы языка Си. Программирование базовых алгоритмических конструкций на языке Си.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	5
4		Подготовка к практическим занятиям: анализ задач, построение алгоритмов	10
5		Оформление отчета по индивидуальному практическому заданию, подготовка к тестированию	5
6	Раздел 3. Типы данных языка Си: скалярные типы, указатели, массивы. Распределение памяти под данные.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	5
7		Подготовка к практическим занятиям: анализ задач, построение алгоритмов	5
8		Оформление отчета по индивидуальному практическому заданию, подготовка к тестированию	20
9	Раздел 4. Вспомогательные алгоритмы. Функции.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	6
10		Подготовка к практическим занятиям: анализ задач, построение алгоритмов	20
11		Оформление отчета по индивидуальному практическому заданию, подготовка к тестированию	10
12		Оформление отчета по индивидуальному практическому заданию, подготовка к тестированию	4
13	Раздел 5. Структурирование данных.	Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	12
14		Подготовка к практическим занятиям: анализ задач, построение алгоритмов	5
15		Оформление отчета по индивидуальному практическому заданию, подготовка к тестированию	3
Всего за 1 семестр			112

4. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

СЕМЕСТР	НЕДЕЛИ СЕМЕСТРА																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1					Отч. по ПЗ, ОС	ДР			Отч. по ПЗ, ОС	ДР			Отч. по ПЗ, ОС	Отч. по ПЗ, ОС		ДР	Отч. по ПЗ, ОС, Вопр. Экз

Условные обозначения:

- ДР – диагностическая работа;
- Отч. по ПЗ – отчет по практическому заданию;
- ОС – устный опрос студентов;
- Вопр. Экз – вопросы к экзамену.

Текущий контроль успеваемости студентов проводится в дискретные временные интервалы в следующих формах:

- диагностическая работа;
- отчет по практическому заданию;
- устный опрос студентов;
- вопросы к экзамену.

Промежуточная аттестация проводится в формах:

- экзамен.

5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

5.1. Основная литература по дисциплине:

1. А. Н. Гуцин, Т. И. Лазарева, И. В. Мартынова. . Типовые алгоритмы и их программирование. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2016, эл. рес.
2. А. Н. Гуцин, Т. И. Лазарева, И. В. Мартынова. . Типовые алгоритмы и их программирование. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2016, 450 экз.
3. А. Н. Гуцин, Т. И. Лазарева, И. В. Мартынова. . Алгоритмы обработки массивов и вспомогательные алгоритмы. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2016, эл. рес.
4. А. Н. Гуцин, Т. И. Лазарева, И. В. Мартынова. . Алгоритмы обработки массивов и вспомогательные алгоритмы. СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2016, 450 экз.
5. Б. В. Керниган, Д. М. Ритчи. . Язык программирования С. М.: Вильямс, 2009, эл. рес.
6. Д. Р. Кувшинов. . Основы программирования. Москва: Юрайт, 2020, эл. рес.
7. И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Фёдоров. . Технологии и методы программирования. Москва: Юрайт, 2021, эл. рес.
8. И. Г. Головин, И. А. Волкова. . Языки и методы программирования. М.: Академия, 2016, 50 экз.
9. И. С. Солдатенко, И. В. Попов. . Практическое введение в язык программирования Си. Санкт-Петербург: Лань, 2022, эл. рес.
10. Н. И. Парфилова, А. В. Пруцков, А. Н. Пылькин. . Информатика и программирование. Основы информатики. М.: Академия, 2016, 250 экз.

5.2. Дополнительная литература по дисциплине:

не требуется.

5.3. Периодические издания:

не требуются.

5.4. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины, электронные библиотечные системы:

1. http://library.voenmeh.ru/jirbis2/index.php?option=com_irbis&view=irbis&Itemid=474 - Электронная библиотека университета — Фундаментальная библиотека БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова;
2. <https://www.codeblocks.org/> - Сайт разработчика свободно распространяемой IDE Code::Blocks;
3. <https://code.visualstudio.com/> - Сайт разработчика свободно распространяемой IDE Visual Studio Code;
4. <https://e.lanbook.com/> - ЭБС ЛАНЬ;
5. <https://urait.ru/> - ЭБС ЮРАЙТ;
6. <https://moodle.voenmeh.ru/course/view.php?id=9542> - Курс "ИНФОРМАТИКА: ОСНОВЫ ПРОГРАММИРОВАНИЯ /О7/Палехова О.А./О731Б, О732Б, О733Б, О734Б, О735Б, О736Б, О737Б, О738Б";
7. https://moodle.voenmeh.ru/pluginfile.php/329384/mod_resource/content/1/gost_19.701-90.pdf - ГОСТ 19.701-90 ЕСПД Схемы алгоритмов, программ, данных и систем;
8. <https://replit.com/languages/c> — C Online Compiler & Interpreter - Replit;
9. <https://www.onlinegdb.com> — GDB online Debugger | Compiler - Code, Compile, Run, Debug online C, C++.

Современные профессиональные базы данных:

1. <https://rusneb.ru> – Национальная электронная библиотека (НЭБ);
2. <https://cyberleninka.ru/> - Научная электронная библиотека «Киберленинка»;
<http://www.rfbr.ru/rffi/ru/library> - Полнотекстовая электронная библиотека Российского фонда фундаментальных исследований.

Информационные справочные системы:

1. Техэксперт – Информационный портал технического регулирования: Нормы, правила, стандарты РФ;

2. http://library.voenmeh.ru/jirbis2/index.php?option=com_irbis&view=irbis&Itemid=457 - БД ГОСТов собственной генерации БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова;
3. <http://www.consultant.ru/> - КонсультантПлюс- информационный портал правовой информации.

5.5. Программное обеспечение:

1. Code::Blocks;
2. Microsoft Visual Studio Community;
3. Kubuntu 18.04 LTS;
4. Офисный пакет Libre Office;
5. Интернет-браузер Mozilla Firefox.

5.6. Информационные технологии:

взаимодействие с обучающимися посредством ЭИОС Moodle БГТУ «ВОЕНМЕХ» им. Д.Ф. Устинова.

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1. Лекционные занятия:

специализированные требования по оборудованию отсутствуют; аудитория с посадочными местами по количеству студентов; доска.

6.2. Практические занятия:

1. Code::Blocks;
2. Microsoft Visual Studio Community;
3. Kubuntu 18.04 LTS;
4. Офисный пакет Libre Office;
5. Интернет-браузер Mozilla Firefox.

6.3. Прочее:

1. рабочее место преподавателя, оснащенное компьютером с доступом в Интернет;
2. рабочие места студентов, оснащенные компьютерами с доступом в Интернет, предназначенные для работы в электронной образовательной среде.

Аннотация рабочей программы

Дисциплина **ИНФОРМАТИКА: ОСНОВЫ ПРОГРАММИРОВАНИЯ** является дисциплиной **обязательной части блока 1** программы подготовки по направлению *09.03.04 Программная инженерия*. Дисциплина реализуется на факультете *О Естественнотехнический БГТУ "ВОЕНМЕХ"* им. Д.Ф. Устинова кафедрой *О7 Информационные системы и программная инженерия*.

Дисциплина нацелена на формирование *компетенций*:

ОПК-2 способность понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности;

ОПК-6 способность разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов.

Содержание дисциплины охватывает круг вопросов, связанных с начальным освоением языка программирования высокого уровня, и включает широкий спектр основных понятий, методов проектирования и программирования, свойств языка программирования. Рассматриваются основные понятия и концепции, наборы символов, ключевые слова, описания и типы переменных, логические выражения, операторы, циклы, основные директивы препроцессора, методики написания и выполнения простейших программ. Обсуждаются вопросы эффективности, переносимости, этапы подготовки, тестирования и отладки программ. Особое внимание уделяется учету характеристик трансляторов, среды программирования и операционных систем, использующихся в настоящее время.

Программой дисциплины предусмотрены следующие **виды контроля**:

Текущий контроль успеваемости студентов проводится в дискретные временные интервалы в следующих формах:

- диагностическая работа;
- отчет по практическому заданию;
- устный опрос студентов;
- вопросы к экзамену.

Промежуточная аттестация проводится в формах:

- экзамен.

Общая трудоемкость освоения дисциплины составляет 5 з.е., **180 ч**. Программой дисциплины предусмотрены лекционные занятия (**34 ч.**), практические занятия (**34 ч.**), самостоятельная работа студента (**112 ч.**).

ТЕХНОЛОГИИ И ФОРМЫ ОБУЧЕНИЯ

Рекомендации по освоению дисциплины для студента

Трудоемкость освоения дисциплины составляет 180 ч., из них 68 ч. аудиторных занятий, и 112 ч., отведенных на самостоятельную работу студента.

Рекомендации по распределению учебного времени по видам самостоятельной работы и разделам дисциплины приведены в таблице.

Контроль освоения дисциплины производится в соответствии с Положением о текущем, рубежном контроле успеваемости и промежуточной аттестации обучающихся.

Формы контроля и критерии оценивания приведены в приложении 3 к Рабочей программе.

Наименование работы	Рекомендуемая литература	Трудоемкость, час.
Раздел 1. Введение.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	И. Г. Головин, И. А. Волкова . . Языки и методы программирования: М.: Академия, 2016 (1) А. Н. Гуцин, Т. И. Лазарева, И. В. Мартынова. . Типовые алгоритмы и их программирование: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2016 (введение, раздел 1) Н. И. Парфилова, А. В. Пруцков, А. Н. Пылькин. . Информатика и программирование. Основы информатики: М.: Академия, 2016 (5) И. С. Солдатенко, И. В. Попов. . Практическое введение в язык программирования Си: Санкт-Петербург: Лань, 2022 (1, 2)	1
Выбор системы программирования и установка ее на домашнем компьютере	И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Фёдоров. . Технологии и методы программирования: Москва: Юрайт, 2021 (1, 2) А. Н. Гуцин, Т. И. Лазарева, И. В. Мартынова. . Типовые алгоритмы и их программирование: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2016 (введение, раздел 1)	1
Итого по разделу 1		2
Раздел 2. Основы языка Си. Программирование базовых алгоритмических конструкций на языке Си.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	Б. В. Керниган, Д. М. Ритчи. . Язык программирования С: М.: Вильямс, 2009 (2, А1, А7, §§ 7.2, 7.4, А4.2, А4.4, 6.1-6.6, гл.3, А9, В4 и §2.11) И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Фёдоров. . Технологии и методы программирования: Москва: Юрайт, 2021 (§§ 5.1-5.3, 5.4, 5.8) Д. Р. Кувшинов. . Основы программирования: Москва: Юрайт, 2020 (1, 2)	5
Подготовка к практическим занятиям: анализ задач, построение алгоритмов	И. С. Солдатенко, И. В. Попов. . Практическое введение в язык программирования Си: Санкт-Петербург: Лань, 2022 (§§3.1, 3.2, 4.1-4.3, 6.1, 6.2) А. Н. Гуцин, Т. И. Лазарева, И. В. Мартынова. . Типовые алгоритмы и их программирование: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2016 (1, 2)	10
Оформление отчета по индивидуальному практическому заданию, подготовка к тестированию	А. Н. Гуцин, Т. И. Лазарева, И. В. Мартынова. . Типовые алгоритмы и их программирование: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2016 (1, 2)	5

Итого по разделу 2		20
Раздел 3. Типы данных языка Си: скалярные типы, указатели, массивы. Распределение памяти под данные.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	Д. Р. Кувшинов. . Основы программирования: Москва: Юрайт, 2020 (§4.2) А. Н. Гуштин, Т. И. Лазарева, И. В. Мартынова. . Алгоритмы обработки массивов и вспомогательные алгоритмы: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2016 (1)	5
Подготовка к практическим занятиям: анализ задач, построение алгоритмов	И. С. Солдатенко, И. В. Попов. . Практическое введение в язык программирования Си: Санкт-Петербург: Лань, 2022 (§§ 4.4, 7.1-7.3, 7.5) И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Фёдоров. . Технологии и методы программирования: Москва: Юрайт, 2021 (§§5.5, 5.6)	5
Оформление отчета по индивидуальному практическому заданию, подготовка к тестированию	А. Н. Гуштин, Т. И. Лазарева, И. В. Мартынова. . Алгоритмы обработки массивов и вспомогательные алгоритмы: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2016 (1) Б. В. Керниган, Д. М. Ритчи. . Язык программирования С: М.: Вильямс, 2009 (5)	20
Итого по разделу 3		30
Раздел 4. Вспомогательные алгоритмы. Функции.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	Б. В. Керниган, Д. М. Ритчи. . Язык программирования С: М.: Вильямс, 2009 (§§1.7, 1.8, 1.10, 4.1, 4.2, 4.3, 4.4, 4.6, 4.10, 5.2, 5.10, 5.11, A11, B7) И. С. Солдатенко, И. В. Попов. . Практическое введение в язык программирования Си: Санкт-Петербург: Лань, 2022 (гл. 5, §7.4)	6
Подготовка к практическим занятиям: анализ задач, построение алгоритмов	И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Фёдоров. . Технологии и методы программирования: Москва: Юрайт, 2021 (§5.7)	20
Оформление отчета по индивидуальному практическому заданию, подготовка к тестированию	Д. Р. Кувшинов. . Основы программирования: Москва: Юрайт, 2020 (§§ 3.1-3.4) А. Н. Гуштин, Т. И. Лазарева, И. В. Мартынова. . Алгоритмы обработки массивов и вспомогательные алгоритмы: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2016 (2)	10
Оформление отчета по индивидуальному практическому заданию, подготовка к тестированию	А. Н. Гуштин, Т. И. Лазарева, И. В. Мартынова. . Алгоритмы обработки массивов и вспомогательные алгоритмы: СПб.БГТУ "ВОЕНМЕХ" им. Д. Ф. Устинова, 2016 (2)	4
Итого по разделу 4		40
Раздел 5. Структурирование данных.		
Изучение предусмотренных программой дидактических единиц по рекомендуемой литературе	И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Фёдоров. . Технологии и методы программирования: Москва: Юрайт, 2021 (§§ 5.9-5.11) Д. Р. Кувшинов. . Основы программирования: Москва: Юрайт, 2020 (5)	12
Подготовка к практическим занятиям: анализ задач, построение алгоритмов	Б. В. Керниган, Д. М. Ритчи. . Язык программирования С: М.: Вильямс, 2009 (§§A2.6, 7.7, 5.5, B1.4, гл.6, B2, B3)	5
Оформление отчета по индивидуальному практическому заданию, подготовка к тестированию	И. С. Солдатенко, И. В. Попов. . Практическое введение в язык программирования Си: Санкт-Петербург: Лань, 2022 (6, 8)	3
Итого по разделу 5		20

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Фонд оценочных средств, позволяющие оценить результаты обучения по данной дисциплине, включают в себя:

- диагностическая работа
- вопросы к экзамену;
- устный опрос студентов;
- отчет по практическому заданию;
- экзамен.

Критерии оценивания

Диагностическая работа

Диагностическая работа проводится в форме теста в ЭИОС Moodle:

- при правильном ответе менее чем на 60% вопросов - не аттестация;
- при правильном ответе на 60% вопросов и более - аттестация.

Вопросы к экзамену

1. Этапы разработки программного обеспечения.
2. Представление информации в компьютере: целые числа, вещественные числа, символы.
3. Представление вещественных чисел в соответствии со стандартом IEEE 754, возможные проблемы вещественных вычислений.
4. Понятие алгоритма, свойства алгоритма.
5. Алгоритмические языки. Основные элементы языков. Синтаксис и семантика языков.
6. Основные элементы языка Си. Структура программы.
7. Классы лексем языка Си.
8. Понятие переменной, константы, операции, выражения в языке Си.
9. Понятие типа данных. Типы данных языка Си.
10. Принципы выбора типов данных при проектировании программы.
11. Объявление переменных в языке Си. Инициализация переменных.
12. Явные и именованные константы в программах на языке Си.
13. Выражения в языке Си, порядок вычисления значений.
14. Правила преобразования типов в языке Си.
15. Ввод/вывод в языке Си.
16. Функции форматного ввода и вывода языка Си.
17. Арифметические операции языка Си.
18. Операции отношения, логические операции языка Си.
19. Операции присваивания языка Си.
20. Поразрядные операции языка Си.
21. Разветвляющиеся алгоритмы. Программирование ветвлений на языке Си: условный оператор, условная операция.
22. Разветвляющиеся алгоритмы. Программирование ветвлений на языке Си. Использование переключателя switch.
23. Циклический алгоритм. Программирование циклов на языке Си. Инструкции while и do...while.
24. Циклический алгоритм. Программирование циклов на языке Си. Инструкция for.
25. Циклический алгоритм. Программирование циклов с предусловием на языке Си.
26. Циклический алгоритм. Способы программирования арифметических циклов на языке Си.
27. Понятие указателя. Объявление указателей на языке Си, инициализация, операции с указателями.
28. Структурный тип данных массив. Массивы в языке Си: индекс массива, инициализация массива. Операции с массивом.
29. Типовые алгоритмы обработки массивов: поиск максимума и минимума.
30. Типовые алгоритмы обработки массивов: вставка и удаление элементов.
31. Типовые алгоритмы обработки массивов. Сортировка.
32. Двумерные и многомерные массивы в языке Си.
33. Массивы и указатели в языке Си.
34. Способы программирования матриц на языке Си.
35. Основные подходы к программированию строк.
36. Массив символов. Си-строка. Способы ввода и вывода строк на языке Си.

37. Стандартные функции языка Си для обработки строк.
38. Принципы нисходящего и восходящего проектирования программ.
39. Функции в языке Си. Прототип функции, определение функции.
40. Функции в языке Си. Вызов функции. Оператор return.
41. Функции в языке Си. Механизм передачи параметров в функцию. Критерии соответствия параметров.
42. Применение указателей в параметрах функций в языке Си.
43. Особенности обработки массивов в функциях при программировании на языке Си.
44. Побочный эффект функции.
45. Рекурсия. Рекурсивные функции в языке Си.
46. Параметры функции main() языка Си.
47. Функции с переменным числом параметров в языке Си.
48. Внутренние и внешние объекты. Время жизни и области видимости объектов.
49. Классы памяти переменных языка Си.
50. Динамическое распределение памяти, функции языка Си для работы с динамической памятью
51. Препроцессор языка Си. Директива #include. Директивы условной компиляции.
52. Препроцессор языка Си. Директива #define.
53. Макросы и функции: сходство и различие.
54. Агрегатные типы данных языка Си: структуры и объединения.
55. Работа с файлами в Си.
56. Поточная обработка текстовых файлов на языке Си.
57. Поточная обработка бинарных файлов на языке Си.
58. Файлы произвольного и последовательного доступа, особенности их обработки.
59. Асимптотические оценки сложности алгоритмов.
60. Оценка вычислительной сложности итерационных алгоритмов.

Устный опрос студентов

Количество вопросов при опросе - не более пяти.

Оценка "отлично" выставляется, если при ответе на вопросы показано глубокое понимание материала.

Оценка "хорошо" выставляется, если показано понимание материала, хотя при ответе на некоторые вопросы допущены ошибки.

Оценка "удовлетворительно" выставляется, если даны правильные ответы хотя бы на половину вопросов.

Перевод оценок за устный опрос в баллы приведен в технологической карте.

Вопросы приведены в файлах с практическими заданиями, а также в УМК дисциплины.

Отчет по практическому заданию

Критерии оценки каждого отчета приведены в ЭИОС moodle, там же находятся шаблоны отчетов.

Экзамен

Обучающийся имеет право на получение оценки "хорошо" или "удовлетворительно" по сумме набранных в течение семестра баллов. Порядок начисления баллов прописан в технологической карте.

Экзаменационный билет включает в себя два теоретических вопроса и задачу на написание программы.

Оценка "отлично" выставляется, если программа работоспособна и написана грамотно, а при ответе на теоретические вопросы показано глубокое понимание материала.

Оценка "хорошо" выставляется, если программа работоспособна, но написана безграмотно, при ответе на теоретические вопросы обнаруживаются пробелы в знаниях.

Оценка "удовлетворительно" выставляется, если программа работоспособна хотя бы на одном наборе данных, но написана безграмотно, при ответе на теоретические вопросы обнаруживаются большие пробелы в знаниях, но обучающийся может более-менее разумно ответить хотя бы на один вопрос билета.

"Не сдано" выставляется, если в течение семестра обучающийся набрал менее 51 балла и за отведенное на проведение экзамена время не смог написать работоспособную даже на одном наборе данных программу.

Паспорт фонда оценочных средств

КУРС	СЕМЕСТР	Наименование разделов и дидактических единиц	ВСЕГО	Аудиторные занятия в контактной форме			Самостоятельная работа студентов	Формируемая компетенция, %		НАИМЕНОВАНИЕ ОЦЕНОЧНОГО СРЕДСТВА
				ВСЕГО	Лекции	Практические занятия		ОПК-2	ОПК-6	
1	1	Раздел 1. Введение.	6	4	2	2	2	5	5	Вопросы к экзамену
1	1	Раздел 2. Основы языка Си. Программирование базовых алгоритмических конструкций на языке Си.	34	14	8	6	20	20	20	Вопросы к экзамену, Отчет по практическому заданию, Устный опрос студентов
1	1	Раздел 3. Типы данных языка Си: скалярные типы, указатели, массивы. Распределение памяти под данные.	44	14	6	8	30	30	15	Вопросы к экзамену, Отчет по практическому заданию, Устный опрос студентов
1	1	Раздел 4. Вспомогательные алгоритмы. Функции.	64	24	12	12	40	20	40	Вопросы к экзамену, Отчет по практическому заданию, Устный опрос студентов
1	1	Раздел 5. Структурирование данных.	32	12	6	6	20	25	20	Вопросы к экзамену, Отчет по практическому заданию, Устный опрос студентов
Всего за 1 семестр			180	68	34	34	112	100	100	
Всего по дисциплине			180	68	34	34	112	100	100	

Критерии оценивания

ОПК-2

Вопросы открытого типа:

- № 1 Что в программировании понимают под выражением?
- № 2 Продолжите фразу: "Область видимости переменной, объявленной внутри блока, распространяется..."
- № 3 Переменная, значением которой является адрес некоторого объекта в оперативной памяти – это _____
- № 4 Как называется символ, ограничивающий Си-строку?
- № 5 Как называется международный стандарт представления действительных чисел в памяти компьютера?
- № 6 Как в языке Си обозначаются комментарии?
- № 7 Что такое семантика языка программирования?
- № 8 Как корректно сравнить на равенство два вещественных числа формата с плавающей точкой?
- № 9 Как называется одно выполнение (проход) цикла?
- № 10 В программе на языке Си имеется объявление двумерного массива

double a[M][N];

где M и N - константы, большие 1. Каковы будут последствия выполнения инструкции

a[M][N]=1;

Вопросы закрытого типа:

- № 1 Что означает следующая фраза: «алгоритм X асимптотически более эффективен, чем Y»?
- X будет лучшим выбором для всех входов
 - X будет лучшим выбором для всех входов, кроме больших входов
 - X будет лучшим выбором для всех входов, за исключением, возможно, небольших входов
 - Y будет лучшим выбором
- № 2 Что понимают под тестированием программы?
- Процесс испытания программного продукта, имеющий своей целью проверку соответствия между реальным поведением программы и её ожидаемым поведением
 - Процесс поиска и выявления ошибок в программе
 - Конкретный вариант значений исходных данных, для которого известен ожидаемый результат
 - Проверку программы на компилируемость
- № 3 Поставьте в соответствие изменению объема памяти, отводимого под число с плавающей точкой, изменение характеристик числа.
- 1) Если увеличится количество разрядов, отводимых под порядок числа
2) Если увеличится количество разрядов, отводимых под мантиссу числа
- А) Увеличится максимальное количество цифр значащей части числа, представимого в этом формате.
Б) Увеличится точность представления данных
В) Увеличится диапазон представления данных
Г) Увеличится максимальное количество цифр дробной части числа, представимого в этом формате
- № 4 Что такое среда разработки программного обеспечения?
- Программа, предназначенная для написания кода программ
 - Программа, предназначенная для запуска других программ
 - Компилятор кода

- Система программных средств, используемая для разработки программного обеспечения
- № 5 Укажите два наилучших алгоритма по критерию трудоемкости
- Алгоритм с линейно-логарифмической скоростью роста
 - Алгоритм с линейной скоростью роста
 - Алгоритм с логарифмической скоростью роста
 - Алгоритм с квадратичной скоростью роста
- № 6 Какой символ или набор символов используется для обозначения шестнадцатеричной константы в тексте программы на языке Си?
- 0 в начале константы
 - 0x в начале константы
 - 0h_ в начале константы
 - h в конце константы
- № 7 Какое минимальное количество циклов нужно организовать для поиска максимального значения элементов прямоугольной матрицы?
- Ни одного
 - 2
 - Зависит от способа организации матрицы в оперативной памяти
 - 1
- № 8 Какого объекта не может существовать в программе на языке Си?
- массива указателей
 - указателя на выражение
 - указателя на функцию
 - указателя на массив
- № 9 В каком случае при присваивании не может произойти потеря данных?
- При присваивании переменной типа int значения типа float
 - При присваивании переменной типа unsigned long int значения типа int
 - При присваивании переменной типа float значения типа int
 - При присваивании переменной типа short int значения типа unsigned char
- № 10 Выберите верные утверждение, касающиеся арифметических преобразований.
- При выполнении арифметических преобразований тип левого операнда в выражении приводится к типу правого операнда.
 - Арифметические преобразования производятся в случаях, когда в выражении не совпадают типы операндов.
 - Арифметические преобразования выполняются неявно.
 - Арифметическими преобразованиями называются преобразования типов в выражениях, не содержащих операции присваивания.

ОПК-6

- Вопросы открытого типа:*
- № 1 Что такое алгоритм?
- № 2 Дан фрагмент программы на языке Си:
- ```
for (x = 0, i = 1 ; i < N ; i++)
 x += i;
```
- Запишите ту же последовательность действий с помощью инструкции *while*.
- № 3                      Сколько элементов массива гарантированно встает на место в результате одного прохода по массиву в алгоритме сортировки пузырьком?
- № 4                      Матрица размера 1000x1200 программируется динамической матрицей с выделением памяти под данные по строкам. Сколько блоков памяти должно быть выделено?

- № 5 Какое минимальное количество циклов потребуется, чтобы найти в массиве минимальное значение и определить, сколько раз оно в массиве встречается?
- № 6 Имеется текстовый файл, содержащий текст романа Л.Н.Толстого «Война и мир». Некто решил пошутить и заменить имя главной героини, назвав ее Анастасией. Опишите на естественном языке алгоритм его действий.
- № 7 Имеется бинарный файл, содержащий вещественные числа формата double. Нужно обнулить 10-е число в файле. Опишите на естественном языке, как это сделать.
- № 8 Имеется объявление типа:

```
struct product
{
 char name [32];
 char material [16];
 struct date
 {
 char day, month;
 short year;
 } date_production;
 char type;
 union option
 {
 int count;
 double volume;
 char nick [16];
 } option;
};
```

- № 9 Сколько компонент содержит этот тип?
- № 10 Какой объем памяти выделяется под переменную структурного типа?
- Программист хочет сравнить на равенство три целые переменные x,y,z и написал фрагмент кода:

```
if (ПРОПУСК) printf("Равны!");
```

Какое логическое выражение может быть на месте ПРОПУСКА? В строку ответа запишите выражение без пробелов, скобки можно использовать только для изменения порядка действий при вычислении выражения.

*Вопросы закрытого типа:*

- № 1 Выберите из перечисленного свойства алгоритма.
- Понятность
  - Функциональность
  - Детерминированность
  - Дискретность
- № 2 Что понимают под тестированием программы?
- Процесс испытания программного продукта, имеющий своей целью проверку соответствия между реальным поведением программы и её ожидаемым поведением
  - Процесс поиска и выявления ошибок в программе
  - Конкретный вариант значений исходных данных, для которого известен ожидаемый результат
  - Проверку программы на компилируемость
- № 3 Что определяет тип данных?
- множество допустимых операций
  - множество допустимых букв
  - размер отводимой памяти
  - формат представления в памяти
  - количество цифр
  - местоположение в памяти

- множество допустимых смысловых конструкций
- множество допустимых значений

- № 4 Что такое побочный эффект функции?
- Изменение функцией значений только глобальных переменных
  - Выполнение функцией дополнительных действий, не предусмотренных алгоритмом, который она реализует
  - Изменение функцией содержимого любых блоков памяти, кроме блоков, занимаемых объектами, принадлежащими этой функции
  - Ситуация, при которой выполнение функции влечет неожиданные последствия для всей программы.
- № 5 Укажите критерии соответствия фактических и формальных параметров функций в языке Си
- по типу
  - по имени
  - по количеству
  - по смыслу
- № 6 Что такое *LVALUE*?
- Предельное значение (limit value)
  - Левый операнд любой бинарной операции
  - Выражение, обозначающее область памяти, в которую можно поместить значение
  - Ошибочное ("левое") значение
- № 7 Двумерный массив `double m[M][N]`; используется для хранения по строкам прямоугольной матрицы размера  $M \times N$ , где  $M \geq 4$ . Какой из приведенных ниже фрагментов позволит вывести на экран первые четыре строки матрицы?

```
for (i = 0 ; i < 4 ; i++)
{
 for (j = 0 ; j < M ; j++)
 printf ("%8.2f", m[i][j]);
 printf ("\n");
}
```

```
for (j = 0 ; j < 4 ; j++)
{
 for (i = 0 ; i < 4 ; i++)
 printf ("%8.2f", m[j][i]);
 printf ("\n");
}
```

```
for (j = 0 ; j < 4 ; j++)
{
 for (i = 0 ; i < N ; i++)
 printf ("%8.2f", m[j][i]);
 printf ("\n");
}
```

```
for (j = 0 ; j < 4 ; j++)
{
```

```

for (i = 0 ; i < N ; i++)
 printf ("%8.2f", m[i][j]);
printf ("\n");
}

```

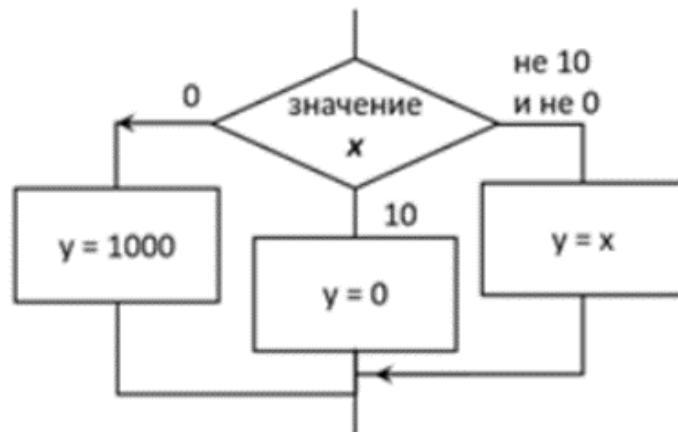
№ 8 Язык Си. Выберите все строки, в которых объявляется указатель на функцию :

- void \* f (int);
- int \* (\*f) (int \*);
- int (\*f) [N];
- void (\*f) (void \*);
- void (\*f) (int);
- void \* f (void \*);
- int \* f (void \*);
- int \* (\*f) [N];

№ 9 Язык Си. Как сохранить значение локальной переменной функции до следующего вызова этой функции?

- Сохранить значение локальной переменной функции до следующего вызова невозможно, так как все локальные переменные после завершения работы функции из памяти удаляются.
- Объявить эту переменную как переменную класса extern.
- Объявить эту переменную как переменную класса static.
- Значения всех переменных, используемых внутри функций, сохраняются в памяти автоматически, никаких действий по сохранению значений предпринимать не надо.

№ 10 Имеется фрагмент алгоритма:



Какими способами можно запрограммировать этот фрагмент на языке Си?

- с помощью инструкции **if-else**
- с помощью операции **?:**
- с помощью инструкции **switch**
- с помощью инструкции **case**