

Смоленский колледж телекоммуникаций (филиал)
федерального государственного бюджетного образовательного учреждения
высшего образования
«Санкт-Петербургский государственный университет телекоммуникаций
им. проф. М.А. Бонч-Бруевича»

УТВЕРЖДАЮ
Заместитель директора по
учебной работе
 И. В. Иванешко
«28» 06 2024 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

ОПЦ.04. Основы алгоритмизации и программирования

общепрофессионального цикла
программы подготовки специалистов среднего звена
по специальности 09.02.07 Информационные системы и программирование

Смоленск, 2024 г.

РАССМОТРЕНО
на заседании методической комиссии
гуманитарных и программно-вычислительных
дисциплин

Председатель Овчинникова И.А.

Протокол № 12

«28» 06 2024 г.

Согласовано
Начальник отдела эксплуатации и внедрения
информационных систем
ОГАУЗ СОМИАЦ
Я.А.Комиссаров
«28» 06 2024 г.

Составитель: Мохнач О.А. – преподаватель высшей квалификационной категории СКТ(ф)СПбГУТ

Рецензенты:

Внутренний рецензент:
Овчинникова И.А., преподаватель высшей квалификационной категории СКТ(ф)СПбГУТ

Внешний рецензент:

Мальцев Е.А. – Ведущий ИТ-специалист ООО ГРАН.

Рабочая программа разработана на основе Федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.07 Информационные системы и программирование, утвержденного приказом Министерства образования и науки РФ № 81547 от 9 декабря 2016 г, профстандарта 06.001 «Программист» (утв. Приказом Министерства труда и социальной защиты РФ от 18 ноября 2013 года №679Н, зарегистрирован Министерством юстиции РФ 18.12.2013, рег.№30635).

Смоленский колледж телекоммуникаций (филиал)
федерального государственного бюджетного образовательного учреждения
высшего образования
«Санкт-Петербургский государственный университет телекоммуникаций
им. проф. М.А. Бонч-Бруевича»

Рецензия
на рабочую программу по дисциплине «Основы алгоритмизации и
программирования»
для специальности 09.02.07 «Информационные системы и программирование»
(автор — Мохнач О.А.)

Представленная к рассмотрению рабочая программа дисциплины рассчитана на 142 часа аудиторных занятий, из которых 76 часов составляют практические занятия. На самостоятельную работу отводится 34 часа.

Рабочая программа учебной дисциплины является частью ППССЗ СКТ(ф)СПбГУТ в соответствии с ФГОС СПО по специальности 09.02.07 Информационные системы и программирование.

Рабочая программа составлена для очной формы обучения. Промежуточная аттестация предусмотрена в форме экзамена.

Программа состоит из 4-х разделов:

Раздел 1. Основные понятия алгоритмизации

Раздел 2. Средства программирования.

Раздел 3. Язык программирования.

Раздел 4. Введение в анализ алгоритмов.

В первом разделе рассмотрены базовые понятия алгоритмизации.

Во втором разделе рассматриваются общие сведения о средствах программирования, жизненном цикле программ, методах и технологиях программирования.

Третий раздел направлен на приобретение знаний об основных принципах структурного и объектно-ориентированного программирования, о базовых конструкциях языка программирования. Подробно рассмотрены вопросы составления выражений, типы данных языка, проработка всех базовых алгоритмических конструкций, применение указателей в различных структурах данных, применение классов и объектов. Уделено внимание использованию библиотеке шаблонов.

Четвертый раздел формирует знания и умения оценки эффективности алгоритмов.

В качестве вариативных умений и знаний введены темы об отладке программ и об обработке исключений и ошибок, об оптимизации кода.

Рабочая программа выполнена на высоком профессиональном уровне, что может свидетельствовать о хорошей подготовке ее разработчика, наличии у него глубоких знаний дисциплины и умении грамотно излагать учебный материал. Присутствие в программе практических занятий позволит хорошо закрепить полученные теоретические знания и получить практический опыт в составлении спецификаций и написании кода, что хорошо скажется на уровне профессиональной подготовки студентов.

В целом представленная программа может быть рекомендована для успешного изучения курса «Основы алгоритмизации и программирования» для специальности 09.02.07 «Информационные системы и программирование».

Рецензент:



Овчинникова И.А.

ООО Контакт-центр ГРАН

Рецензия

на рабочую программу по дисциплине «Основы алгоритмизации и программирования»

для специальности 09.02.07 «Информационные системы и программирование»

(автор — Мохнач О.А.)

Рабочая программа дисциплины составлена в соответствии с требованиями федерального государственного образовательного стандарта к минимуму содержания и уровню подготовки выпускников по дисциплине ОПЦ.04 «Основы алгоритмизации и программирования» ФГОС. Рабочая программа предназначена для образовательных организаций СПО, ведущих подготовку специалистов технического профиля и является частью программы подготовки специалистов среднего звена, разработанных в соответствии с ФГОС СПО по специальности: 09.02.07 – Информационные системы и программирование.

Программа рассчитана на 142 аудиторных часа, из них 76 часов отводится на выполнение практических занятий, что соответствует учебному плану колледжа по специальности 09.02.07 «Информационные системы и программирование».

В программу входят следующие разделы:

Раздел 1. Основные понятия алгоритмизации

Раздел 2. Средства программирования.

Раздел 3. Язык программирования.

Раздел 4. Введение в анализ алгоритмов.

Изучение разделов программы направлено на формирование у студентов знаний об основных методах построения алгоритмов, типах данных, базовых конструкциях изучаемых языков программирования, принципах структурного и модульного программирования, принципах объектно-ориентированного программирования.

В результате освоения дисциплины обучающиеся должны уметь:

- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на языке программирования.

В программу также включены знания и умения, такие как тестирование и отладка программы, обработка ошибок, культура и стили программирования, использование стандартной библиотеки шаблонов классов, в результате изучения которых и выполнения практических занятий, студенты смогут использовать полученные знания и умения в дальнейшем освоении профессиональных модулей и в практической деятельности.

Освоение данной программы готовит студентов для освоения профессиональных компетенций:

- ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.
- ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием.
- ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств.
- ПК 1.4. Выполнять тестирование программных модулей
- ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода.
- ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.
- ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.

Представленная программа разработана на высоком уровне и может быть рекомендована для изучения курса для специальности 09.02.07 – Информационные системы и программирование.

Рецензент: вед.специалист ООО ГРАН

 Мальцев Е.А.

СОДЕРЖАНИЕ

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	4
2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ	7
3. УСЛОВИЯ РЕАЛИЗАЦИИ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	13
4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ	15
ПРИЛОЖЕНИЕ 1	17
ПРИЛОЖЕНИЕ 2	21
ПРИЛОЖЕНИЕ 3	22

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

ОПЦ.04 Основы алгоритмизации и программирования

1.1 Область применения программы

Рабочая программа учебной дисциплины является частью ППССЗ СКТ(ф)СПбГУТ в соответствии с ФГОС СПО по специальности 09.02.07 Информационные системы и программирование.

Рабочая программа составлена для очной формы обучения.

1.2. Место дисциплины в структуре программы подготовки специалистов среднего звена: дисциплина относится к общепрофессиональным дисциплинам цикла профессиональной подготовки ППССЗ.

1.3. Цели и задачи дисциплины – требования к результатам освоения дисциплины:

Обязательная часть

В результате освоения дисциплины студент должен уметь:

- У1. Разрабатывать алгоритмы для конкретных задач.
- У2. Использовать программы для графического отображения алгоритмов.
- У3. Работать в среде программирования.
- У4. Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.
- У5. Оформлять код программы в соответствии со стандартом кодирования.
- У6. Выполнять тестирование, отладку кода программы.

В результате освоения дисциплины студент должен знать:

- 31. Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции. Этапы решения задачи на компьютере;
- 32. Эволюция языков программирования, их классификация, понятие системы программирования.
- 33. Основные элементы языка программирования: простые и структурированные типы данных, структуру программы, операторы и операции, управляющие структуры, файлы, указатели.
- 34. Основные принципы структурного программирования. Подпрограммы, составление библиотек подпрограмм.
- 35. Стили и культура программирования. Стандарты кодирования. Рефакторинг и оптимизация кода.
- 36. Объектно-ориентированная модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка.

Содержание дисциплины ориентировано на подготовку студентов к освоению профессиональных модулей ППССЗ по специальности 09.02.07 Информационные системы и программирование и овладению профессиональными компетенциями:

- ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.
- ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием.
- ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств.
- ПК 1.4. Выполнять тестирование программных модулей
- ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода.
- ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.
- ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.

В процессе освоения дисциплины у студентов должны быть сформированы общие компетенции:

ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам.

ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности.

ОК 04. Эффективно взаимодействовать и работать в коллективе и команде

ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста.

ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках.

Вариативная часть

С целью удовлетворения запросов рынка труда и удовлетворения требований профессионального стандарта (согласно Приказу Министерства труда и социальной защиты РФ от 18 ноября 2013 г. N 679н «Об утверждении профессионального стандарта "Программист"») студент должен

уметь:

У7. Определять сложность работы алгоритмов.

У8. Обрабатывать ошибки и исключительные ситуации;

У9. Применять объекты готовых шаблонов классов.

У10. Проводить рефакторинг и оптимизацию программного кода.

знать:

37. Структуры данных: стек, очередь, бинарное дерево, списки, множества и пр.

38. Методы построения алгоритмов. Алгоритмы сортировки и поиска.

39. Введение в оценку сложности алгоритма.
310. Библиотеки шаблонов и классов;
311. Исключительные ситуации и ошибки;
312. Базовые понятия отладки и тестирования.

1.4. Количество часов на освоение программы учебной дисциплины:

максимальной учебной нагрузки - 184 часа, из них - 138 часов обязательная часть, 46 часов – вариативная часть, в том числе:

- обязательной аудиторной учебной нагрузки студента - 142 часов, из них:
Практических занятий - 76 часов;
- самостоятельной работы студента - 34 часа;
- промежуточной аттестации – 6 часов;
- консультации - 2 часа.

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Виды учебной работы	Объем часов
Максимальная учебная нагрузка (всего)	184
Обязательная аудиторная учебная нагрузка (всего)	142
в том числе: лекции, уроки	66
Лабораторные занятия	-
практические занятия	76
Контрольная работа	-
Курсовое проектирование	-
Самостоятельная работа студента	34
в том числе:	34
Домашняя работа по подготовке ответов на контрольные вопросы, работа с основной и дополнительной литературой, выполнение индивидуальных и групповых заданий	
Консультации	2
Промежуточная аттестация в форме экзамена	6

2.2. Тематический план и содержание учебной дисциплины ОПЦ.04 Основы алгоритмизации и программирования

Наименование разделов и тем	Содержание учебного материала, практические занятия, самостоятельная работа студентов	Объем в часах
Раздел 1.	Основные понятия алгоритмизации	
Тема 1.1. Основные понятия алгоритмизации	<p>Содержание учебного материала:</p> <p>1. Алгоритмы. Общие сведения. Свойства алгоритмов. Базовые алгоритмические конструкции.</p> <p>2. Способы представления алгоритмов. Основы решения алгоритмической задачи. Реализация алгоритмов средствами автоматизированного проектирования.</p> <p>Практические занятия:</p> <p>1. Составление ветвящегося алгоритма.</p> <p>2. Составление циклических алгоритмов.</p> <p>Самостоятельная работа студентов:</p> <p>Проработка конспекта, дополнительной литературы, поиск информации по заданным темам; подготовка к тестированию.</p> <p>Выполнение заданий: Составление алгоритмов.</p>	4
Раздел 2.	Средства программирования	
Тема 2.1. Средства и технологии программирования	<p>Содержание учебного материала:</p> <p>1. Обзор языков программирования. Области применения языков программирования. Среда проектирования. Компиляторы и интерпретаторы. Жизненный цикл программы. Основные этапы решения задач на компьютере.</p> <p>2. Современные технологии программирования. Структурное программирование.</p> <p>Самостоятельная работа студентов:</p> <p>Проработка конспекта, дополнительной литературы, поиск информации по заданным темам; подготовка к письменному опросу.</p> <p>Эволюция языков программирования: создание презентаций.</p>	4
Раздел 3.	Язык программирования.	
Тема 3.1. Основы языка программирования.	<p>Содержание учебного материала:</p> <p>1. Лексические основы языка. Структура программы. Идентификаторы.</p> <p>2. Типы данных. Скалярные типы данных. Объявление переменных. Виды выражений.</p> <p>3. Правила формирования и вычисления выражений. Преобразование типов.</p> <p>4. Операторы общего назначения. Оператор присваивания. Ввод и вывод данных. Простой и составной оператор. Условный оператор. Оператор выбора.</p> <p>5. Цикл с предусловием. Цикл с постусловием. Цикл с параметром. Вложенные циклы. Досрочный выход из цикла.</p> <p>6. Стили программирования. Культура программного кода. Стандарты кодирования.</p> <p>Практические занятия:</p> <p>3. Составление линейной программы.</p>	12
		12

	<p>4. Составление ветвящейся программы.</p> <p>5. Реализация множественного ветвления.</p> <p>6. Реализация цикла с параметром.</p> <p>7. Реализация условных циклов.</p> <p>8. Реализация вложенных циклов.</p>	
	<p>Самостоятельная работа студентов:</p> <p>Подготовка докладов, презентаций</p> <p>Проработка конспекта, основной и дополнительной литературы, поиск информации по заданным темам; подготовка к защите практических занятий, ответы на контрольные вопросы; подготовка к тестированию</p> <p>Выполнение индивидуальных заданий.</p>	4
Тема 3.2. Структурированные типы данных.	<p>Содержание учебного материала:</p> <p>1. Массивы, определение массивов. Способы заполнения массивов. Обработка массивов. Сортировка массивов методом прямого выбора и пузырька.</p> <p>2. Символы и строки. Обработка строк в стиле С. Строки как элементы класса string.</p> <p>3. Структурированный тип данных – множество. Комбинированный тип данных – запись (структура).</p> <p>Практические занятия:</p> <p>9. Обработка одномерных массивов.</p> <p>10. Сортировка массивов методом грубой силы.</p> <p>11. Обработка двумерных массивов.</p> <p>12. Обработка символов. Обработка строк в стиле С.</p> <p>13. Обработка строк как элементы класса string.</p> <p>14. Реализация структур.</p>	6
	<p>Самостоятельная работа студентов:</p> <p>Подготовка докладов, презентаций</p> <p>Проработка конспекта, основной и дополнительной литературы, поиск информации по заданным темам; подготовка к защите практических занятий, ответы на контрольные вопросы; подготовка к тестированию</p> <p>Выполнение индивидуальных заданий.</p>	12
Тема 3.3. Отладка и тестирование программ.	<p>Содержание учебного материала:</p> <p>1. Отладка и тестирование. Методы отладки. Использование отладчика. Примеры построения тестовых сценариев.</p> <p>Практические занятия:</p> <p>15. Тестирование и отладка программы.</p> <p>Самостоятельная работа студентов:</p> <p>Проработка конспекта, дополнительной литературы, поиск информации по заданным темам; подготовка к защите практических занятий, ответы на контрольные вопросы; подготовка к тестированию</p> <p>Выполнение тестирования и отладки программного продукта с протоколированием результатов.</p>	5

Тема 3.4. Подпрограммы	Содержание учебного материала: 1. Подпрограммы. Преимущества использования подпрограмм. Структура подпрограммы. Вызов подпрограммы. Стандартные функции. Написание пользовательских функций. Формальные и фактические параметры. Механизм передачи параметров. 2. Перегрузка функций. Функции с параметром по умолчанию. Рекурсивные функции. 3. Алгоритмы сортировки слиянием и быстрой сортировки. Алгоритмы предварительной сортировки. Алгоритм бинарного поиска.	6
	Практические занятия: 16. Реализация пользовательских функций. 17. Перегрузка функций. 18. Рекурсивные функции. 19. Применение рекурсии в обработке массивов и строк.	8
	Самостоятельная работа студентов: Проработка конспекта, основной и дополнительной литературы, поиск информации по заданным темам; подготовка к защите практических занятий, ответы на контрольные вопросы; подготовка к тестированию Реализация задачи: рекурсивное определение функций.	4
	Содержание учебного материала: 1. Файловый ввод-вывод. Текстовые файлы. Бинарные файлы.	2
Тема 3.5. Работа с файлами	Практические занятия: 20. Работа с текстовыми файлами. 21. Работа с бинарными файлами.	4
	Самостоятельная работа студентов: Проработка конспекта, дополнительной литературы, поиск информации по заданным темам; подготовка к защите практических занятий, ответы на контрольные вопросы; подготовка к тестированию.	2
	Содержание учебного материала: 1. Ссылки и указатели. Массивы и указатели. Связь строк и указателей. 2. Структуры данных. Списки. Структуры данных: стек и очередь. Структуры данных: двоичные деревья 3. Реализация динамических структур данных. Задача о стеке.	6
	Практические занятия: 22. Использование указателей для обработки массивов. 23. Использование указателей для обработки строк. 24. Реализация стеков и очередей.	6
Тема 3.6. Указатели. Структуры данных.	Самостоятельная работа студентов: Проработка конспекта, дополнительной литературы, поиск информации по заданным темам; подготовка к защите практических занятий, ответы на контрольные вопросы; подготовка к тестированию Выполнение групповых заданий	4

Тема 3.7. Модули и библиотеки.	Содержание учебного материала: 1.Модульное программирование. Создание собственной библиотеки. Подключение библиотек.	2
	Практические занятия: 25. Создание собственной библиотеки.	2
	Самостоятельная работа для студентов: Создание презентаций, докладов; Проработка конспекта, основной и дополнительной литературы, поиск информации по заданным темам; подготовка к защите практических занятий, ответы на контрольные вопросы; подготовка к тестированию Выполнение заданий: Добавление в библиотеку собственной функции.	1
Тема 3.8. Объектно- ориентированн ое программирова ние.	Содержание учебного материала: 1.Принципы ООП. Определение классов и их методов. Управление доступом к членам класса. Конструкторы и деструкторы. Порядок вызова 2.Друзья классов. Статический полиморфизм. Перегрузка операций. 3.Виды отношений между классами. Наследование. Динамический полиморфизм. Механизм виртуальных функций. 4.Параметрический полиморфизм. Шаблоны функций. Шаблонные классы. 5.Стандартная библиотека шаблонов. Контейнеры. Обзор контейнеров. Итераторы. 6.Алгоритмы в контейнерах. Достоинства и недостатки STL-подхода. 7.Контейнер-вектор. Контейнер-список. Примеры решения задач с использованием контейнеров STL.	14
	Практические занятия: 26. Определение классов и описание их методов 27. Использование конструктора. 28. Перегрузка бинарных операций. 29. Перегрузка унарных операций. 30. Использование свойства наследования. 31. Реализация виртуальных функций. 32. Создание шаблона функции. 33. Реализация контейнера-вектора. 34. Реализация контейнера-списка.	18
	Самостоятельная работа для студентов: Проработка конспекта, основной и дополнительной литературы, поиск информации по заданным темам; подготовка к защите практических занятий, ответы на контрольные вопросы; подготовка к тестированию Выполнение индивидуальных заданий: Написание объектно-ориентированного приложения Выполнение групповых заданий: Реализация контейнеров	5
Тема 3.9. Обработка	Содержание учебного материала: 1.Понятие исключительной ситуации. Обработка исключительных ситуаций.	2

исключительных ситуаций и ошибок.	Практические занятия: 35. Обработка исключений.	2
	Самостоятельная работа для студентов: Проработка конспекта, основной и дополнительной литературы, поиск информации по заданным темам; подготовка к защите практических занятий, ответы на контрольные вопросы; подготовка к тестированию Выполнение индивидуальных заданий.	1
Тема 3.10. Оптимизация и рефакторинг.	Содержание учебного материала: 1.Оптимизация кода. Примеры. Рефакторинг кода.	2
	Практические занятия: 36. Рефакторинг и оптимизация. Соответствие стандартам кодирования.	2
	Самостоятельная работа студентов: Проработка конспекта, дополнительной литературы, поиск информации по заданным темам; подготовка к защите практических занятий, ответы на контрольные вопросы; подготовка к тестированию Выполнение индивидуальных заданий: Оценка сложности алгоритма.	1
Раздел 4.	Введение в анализ алгоритмов.	
Тема 4.1. Методы вычисления сложности алгоритмов.	Содержание учебного материала: 1.Сравнительная оценка алгоритмов. Асимптотический анализ функции трудоемкости и сложность алгоритма. Классы сложности алгоритмов. 2.Пример полного анализа алгоритма решения задачи о сумме. Анализ различных алгоритмов сортировки. Рекурсивные алгоритмы и методы их анализа.	4
	Практические занятия: 37. Расчет функции трудоемкости нерекурсивного алгоритма. 38. Расчет функции трудоемкости рекурсивного алгоритма.	4
	Самостоятельная работа студентов: Проработка конспекта, дополнительной литературы, поиск информации по заданным темам; подготовка к защите практических занятий, ответы на контрольные вопросы; подготовка к тестированию Выполнение заданий: Оценка сложности и трудоемкости алгоритма.	3
	Консультации	2
Промежуточная аттестация		6
Всего:		184

3. УСЛОВИЯ РЕАЛИЗАЦИИ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1. Материально – техническое обеспечение

Обучение по программе дисциплины осуществляется в лаборатории «Программирования и баз данных».

Технические средства обучения:

Рабочая станция студента (комплект с двумя мониторами Dell SE2416H 24”, клавиатурой и мышью, процессор IntelPentiumDualCore G4620 3.7 GHz, оперативная память DDR4 16 Gb, жесткие диски SSD KINGSTON 512 Гб , HDD WDC 1TB, видеоадаптер GTX 1050 2 Gb) - 12 шт.

- Рабочая станция преподавателя (комплект с монитором Dell SE2416H 24”, клавиатурой и мышью, процессор IntelCore i5 7400 3.0 GHz, оперативная память DDR4 16 Gb, жесткие диски SSD KINGSTON 512 Гб) - 1 шт.
- Сервер лаборатории (процессор IntelCore i7 7700 3.6 GHz, оперативная память 32 Gb, жесткий диск 2 Tb) - 1 шт.
- Проектор Sanyo.
- Усилитель мощности Crown XLi800.
- Акустическая система Bosch.
- Презентатор Samsung.
- Флипчарт на треноге.
- Принтер А3 цветной Cannon PIXMA iX 6840.

Программные средства обучения:

- схемы и презентации по темам дисциплины;
- виртуальные тесты;
- Операционная система: Windows 10;
- Libre Office 2003, MS Office 2007,
- MicrosoftVisualStudio 2019,
- MinGW, Eclipse, CodeBlocks,
- NetBeans.
- сетевая тестирующая оболочка Айрен (свободное ПО);
- презентации по темам дисциплины;
- виртуальные тесты.

3.2. Информационное обеспечение обучения

Основные источники:

1. Андрианова, А. А. Алгоритмизация и программирование. Практикум: учебное пособие /А. А. Андрианова, Л. Н. Исмагилов, Т. М. Мухтарова. — Санкт-Петербург: Лань, 2022. — ISBN 978-5-8114-3336-0. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/206258/>.
2. Залогова, Л. А. Основы объектно-ориентированного программирования на базе языка C# / Л. А. Залогова. — 4-е изд., стер. — Санкт-Петербург: Лань, 2023. — 192 с. — ISBN 978-5-507-48276-4. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/345992>.
3. Огнева, М. В. Программирование на языке С++: практический курс: учебное пособие для среднего профессионального образования / М. В. Огнева, Е. В. Кудрина. — Москва: Издательство Юрайт, 2020. — 335 с. — (Профессиональное образование). — ISBN 978-5-534-05780-5. — Текст: электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/454231>
4. Павловская Т. А. С/C++. Программирование на языке высокого уровня. — (Серия «Учебник для вузов»). / Т.А. Павловская. - Санкт-Петербург: Питер, 2021. - 464 с. - ISBN 978-5-4461-1350-7. - URL: <https://ibooks.ru/bookshelf/376844/reading> - Текст: электронный.

5. Рысин, М. Л. Введение в структуры и алгоритмы обработки данных: учебное пособие / М. Л. Рысин, М. В. Сартаков, М. Б. Туманова. — Москва: РТУ МИРЭА, 2022 — Часть 1: Сложность алгоритмов. Сортировки. Линейные структуры данных. Поиск в таблице — 2022. — ISBN 978-5-7339-1612-5. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/256592>.

Дополнительные источники:

1. Барков, И. А. Объектно-ориентированное программирование / И. А. Барков. — 2-е изд., стер. — Санкт-Петербург: Лань, 2023. — ISBN 978-5-507-47113-3. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/329549>.
2. Парфенов, Д. В. Промышленное программирование с использованием языка C++: учебное пособие / Д. В. Парфенов, Д. А. Петруевич. — Москва: РТУ МИРЭА, 2023. — 131 с. — ISBN 978-5-7339-1708-5. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/331619>.
3. Плаксин М.А. Тестирование и отладка программ для профессионалов будущих и настоящих —4-е изд., электрон. / М.А. Плаксин. - Москва: Лаборатория знаний, 2020. - 170 с. - ISBN 978-5-00101-810-0. - URL: <https://ibooks.ru/bookshelf/353395/reading> - Текст: электронный.
4. Рацеев, С. М. Программирование. Лабораторный практикум / С. М. Рацеев. — Санкт-Петербург: Лань, 2023. — 104 с. — ISBN 978-5-507-45194-4. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/292907>.
5. Ульянова, Н. Д. Основные принципы алгоритмизации: учебно-методическое пособие / Н. Д. Ульянова. — Брянск: Брянский ГАУ, 2020. — 56 с. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/172114>.
6. Черенкова, И. А. Основы алгоритмизации и программирования на языке высокого уровня: учебное пособие / И. А. Черенкова, О. А. Кишкинова, Ю. Б. Миндлин. — Москва: МГАВМиБ им. К.И. Скрябина, 2022. — ISBN 978-5-4443-0235-4. — Текст: электронный // Лань: электронно-библиотечная система. — URL: <https://e.lanbook.com/book/331409>.

Интернет-ресурсы:

1. CIT-Forum: Центр информационных технологий: материалы сайта [Электронный ресурс]. - Режим доступа: <http://citforum.ru/programming/c.shtml>, свободный.
2. Cplusplus.com - The C++ Resources Network: материалы сайта [Электронный ресурс]. - Режим доступа: <http://wwwcplusplus.com/>, свободный.
3. CppStudio: Основы программирования на языках Си и С++ для начинающих [Электронный ресурс]. - Режим доступа: <http://cppstudio.com/>, свободный
4. CyberForum.ru - форум программистов и систем администраторов [Электронный ресурс]. - Режим доступа: <http://www.cyberforum.ru/cpp-beginners/>, свободный
5. Библиотека учебных курсов/ Интернет-Университет информационных технологий - Интuit (Национальный Открытый университет) [Электронный ресурс]. - Режим доступа: <http://www.intuit.ru/studies/courses/13869/1266/info/>, свободный
6. Онлайн учебники и справочные материалы по С++ [Электронный ресурс]. - Режим доступа: <https://metanit.com/>, свободный.
7. Электронный проект: теория алгоритмов [Электронный ресурс]. - Режим доступа: <http://th-algoritmov.narod.ru/>, свободный.

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Контроль и оценка результатов освоения дисциплины осуществляется преподавателем в процессе проведения практических работ, тестирования, а также выполнения студентами индивидуальных заданий, проектов.

Результаты обучения (освоенные умения, усвоенные знания)	Формы и методы контроля и оценки результатов обучения
уметь:	
У1-Разрабатывать алгоритмы для конкретных задач.	Формализованное наблюдение и оценка результата выполнения и защиты практических занятий №№ 1-2, 37-38. Оценка результатов электронного тестирования; Оценка выполнения самостоятельной работы: презентаций, отчетов; Наблюдение за выполнением конкретных заданий.
У2-Использовать программы для графического отображения алгоритмов.	Формализованное наблюдение и оценка результата выполнения и защиты практических занятий №№ 1-2, 37-38. Оценка результатов электронного тестирования; Наблюдение за выполнением конкретных заданий.
У3- Работать в среде программирования.	Формализованное наблюдение и оценка результата выполнения и защиты практических занятий №№3-36. Оценка результатов электронного тестирования; Оценка выполнения самостоятельной работы. Наблюдение за выполнением конкретных заданий.
У4-Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.	Формализованное наблюдение и оценка результата выполнения и защиты практических занятий №№ 3-14, 16-35. Оценка результатов электронного тестирования; Оценка выполнения самостоятельной работы: отчетов, программных продуктов; Наблюдение за выполнением конкретных заданий.
У 5-Оформлять код программы в соответствии со стандартом кодирования.	Формализованное наблюдение и оценка результата выполнения и защиты практических занятий №№ 9-36. Наблюдение за выполнением конкретных заданий.
У6-Выполнять тестирование, отладку кода программы.	Формализованное наблюдение и оценка результата выполнения и защиты практических занятий №№ 15. Оценка результатов электронного тестирования; Наблюдение за выполнением конкретных заданий.
У7-Определять сложность работы алгоритмов.	Формализованное наблюдение и оценка результата выполнения и защиты практических занятий №№ 37-38. Оценка результатов электронного тестирования; Оценка выполнения самостоятельной работы; Наблюдение за выполнением конкретных заданий.
У8 - Обрабатывать ошибки и исключительные ситуации;	Формализованное наблюдение и оценка результата выполнения и защиты практических занятий №№ 35. Оценка результатов электронного тестирования;

	Наблюдение за выполнением конкретных заданий.
У9 - Применять объекты готовых шаблонов классов.	Формализованное наблюдение и оценка результата выполнения и защиты практических занятий №№ 24, 32-34. Оценка результатов электронного тестирования; Оценка выполнения самостоятельной работы. Наблюдение за выполнением конкретных заданий.
У10 - Проводить рефакторинг и оптимизацию программного кода.	Формализованное наблюдение и оценка результата выполнения и защиты практических занятий №№ 36. Наблюдение за выполнением конкретных заданий.
<i>знать:</i>	
31 - Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции. Этапы решения задачи на компьютере;	Оценка точности воспроизведения изученной информации, анализа и сравнения данных в ходе выполнения самостоятельных работ, индивидуальных заданий, тестирования, устных опросов, выполнения программных продуктов по темам 1.1-4.1.
32 - Эволюция языков программирования, их классификация, понятие системы программирования.	
33 - Основные элементы языка программирования: простые и структурированные типы данных, структуру программы, операторы и операции, управляющие структуры, файлы, указатели.	
34 - Основные принципы структурного программирования. Подпрограммы, составление библиотек подпрограмм.	
35 - Стили и культура программирования. Стандарты кодирования. Рефакторинг и оптимизация кода.	
36 – Объектно-ориентированная модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка.	
37 - Структуры данных: стек, очередь, бинарное дерево, списки, множества и пр	
38 - Методы построения алгоритмов. Алгоритмы сортировки и поиска.	
39 - Введение в оценку сложности алгоритма.	
310 - Библиотеки шаблонов и классов;	
311 -Исключительные ситуации и ошибки;	
312 - Базовые понятия отладки и тестирования.	

Конкретизация результатов освоения дисциплины

ПК 1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.	
Уметь: У1-Разрабатывать алгоритмы для конкретных задач. У2-Использовать программы для графического отображения алгоритмов. У7-Определять сложность работы алгоритмов.	Тематика лабораторных/практических занятий: ПЗ 1 Составление ветвящегося алгоритма. ПЗ 2 Составление циклических алгоритмов. ПЗ 37 Расчет функции трудоемкости нерекурсивного алгоритма. ПЗ 38 Расчет функции трудоемкости рекурсивного алгоритма.
Знать: 31 - Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции. Этапы решения задачи на компьютере; 37 - Структуры данных: стек, очередь, бинарное дерево, списки, множества и пр 38 - Методы построения алгоритмов. Алгоритмы сортировки и поиска. 39 - Введение в оценку сложности алгоритма.	Перечень тем: 1.1.1. Алгоритмы. Общие сведения. Свойства алгоритмов. Базовые алгоритмические конструкции. 1.1.2. Способы представления алгоритмов. Основы решения алгоритмической задачи. Реализация алгоритмов средствами автоматизированного проектирования. 3.4.3. Алгоритмы сортировки слиянием и быстрой сортировки. Алгоритмы предварительной сортировки. Алгоритм бинарного поиска. 3.6.2. Структуры данных. Списки. Структуры данных: стек и очередь. Структуры данных: двоичные деревья 3.6.3. Реализация динамических структур данных. Задача о стеке. 4.1.1. Сравнительная оценка алгоритмов. Асимптотический анализ функции трудоемкости и сложность алгоритма. Классы сложности алгоритмов. 4.1.2. Пример полного анализа алгоритма решения задачи о сумме. Анализ различных алгоритмов сортировки. Рекурсивные алгоритмы и методы их анализа.
Самостоятельная работа	Тематика самостоятельной работы: Проработка конспекта, дополнительной литературы, поиск информации по заданным темам; подготовка к защите практических занятий, ответы на контрольные вопросы; подготовка к тестированию Выполнение самостоятельной работы, индивидуальных и групповых заданий.
ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием.	
Уметь: У3- Работать в среде программирования. У4-Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования. У8 - Обрабатывать ошибки и исключительные ситуации; У9 - Применять объекты готовых шаблонов классов.	Тематика лабораторных/практических занятий: ПЗ 3. Составление линейной программы. ПЗ 4. Составление ветвящейся программы. ПЗ 5. Реализация множественного ветвления. ПЗ 6. Реализация цикла с параметром. ПЗ 7. Реализация условных циклов. ПЗ 8. Реализация вложенных циклов. ПЗ 9. Обработка одномерных массивов. ПЗ 10. Сортировка массивов методом грубой силы. ПЗ 11. Обработка двумерных массивов. ПЗ 12. Обработка символов. Обработка строк в стиле С. ПЗ 13. Обработка строк как элементы класса string.

	<p>ПЗ 14. Реализация структур.</p> <p>ПЗ 16. Реализация пользовательских функций.</p> <p>ПЗ 17. Перегрузка функций.</p> <p>ПЗ 18. Рекурсивные функции.</p> <p>ПЗ 19. Применение рекурсии в обработке массивов и строк.</p> <p>ПЗ 20. Работа с текстовыми файлами.</p> <p>ПЗ 21. Работа с бинарными файлами.</p> <p>ПЗ 22. Использование указателей для обработки массивов.</p> <p>ПЗ 23. Использование указателей для обработки строк.</p> <p>ПЗ 24. Реализация стеков и очередей.</p> <p>ПЗ 25. Создание собственной библиотеки.</p> <p>ПЗ 26. Определение классов и описание их методов</p> <p>ПЗ 27. Использование конструктора.</p> <p>ПЗ 28. Перегрузка бинарных операций.</p> <p>ПЗ 29. Перегрузка унарных операций.</p> <p>ПЗ 30. Использование свойства наследования.</p> <p>ПЗ 31. Реализация виртуальных функций.</p> <p>ПЗ 32. Создание шаблона функции.</p> <p>ПЗ 33. Реализация контейнера-вектора.</p> <p>ПЗ 34. Реализация контейнера-списка.</p> <p>ПЗ 35. Обработка исключений.</p>
Знать: <p>32 - Эволюция языков программирования, их классификация, понятие системы программирования.</p> <p>33 - Основные элементы языка программирования: простые и структурированные типы данных, структуру программы, операторы и операции, управляющие структуры, файлы, указатели.</p> <p>34 - Основные принципы структурного программирования. Подпрограммы, составление библиотек подпрограмм.</p> <p>36 – Объектно-ориентированная модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка.</p> <p>37 - Структуры данных: стек, очередь, бинарное дерево, списки, множества и пр</p> <p>310 - Библиотеки шаблонов и классов;</p> <p>311 -Исключительные ситуации и ошибки;</p>	Перечень тем: <p>Тема 2.1. Средства и технологии программирования</p> <p>Тема 3.1. Основы языка программирования.</p> <p>Тема 3.2. Структурированные типы данных.</p> <p>Тема 3.4. Подпрограммы</p> <p>Тема 3.5. Работа с файлами</p> <p>Тема 3.6. Указатели. Структуры данных.</p> <p>Тема 3.7. Модули и библиотеки.</p> <p>Тема 3.8. Объектно-ориентированное программирование.</p> <p>Тема 3.9. Обработка исключительных ситуаций и ошибок.</p>
Самостоятельная работа	Тематика самостоятельной работы: <p>Проработка конспекта, дополнительной литературы, поиск информации по заданным темам; подготовка к защите практических занятий, ответы на контрольные вопросы; подготовка к тестированию</p> <p>Выполнение самостоятельной работы, индивидуальных и</p>

	групповых заданий.
ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств.	
Уметь: У6. Выполнять тестирование, отладку кода программы. У8. Обрабатывать ошибки и исключительные ситуации;	Тематика лабораторных/практических занятий: ПЗ 15. Тестирование и отладка программы. ПЗ 35. Обработка исключений.
Знать: 311. Исключительные ситуации и ошибки 312. Базовые понятия отладки и тестирования.	Перечень тем: 1.1.1. Отладка и тестирование. Методы отладки. Использование отладчика. Примеры построения тестовых сценариев. 3.9.1. Понятие исключительной ситуации. Обработка исключительных ситуаций.
Самостоятельная работа	
Тематика самостоятельной работы: Проработка конспекта, дополнительной литературы, поиск информации по заданным темам; подготовка к защите практических занятий, ответы на контрольные вопросы; подготовка к тестированию Выполнение тестирования и отладки программного продукта с протоколированием результатов.	
ПК 1.4. Выполнять тестирование программных модулей	
Уметь: У6. Выполнять тестирование, отладку кода программы. У8. Обрабатывать ошибки и исключительные ситуации;	Тематика лабораторных/практических занятий: ПЗ 15. Тестирование и отладка программы. ПЗ 35. Обработка исключений.
Знать: 311. Исключительные ситуации и ошибки 312. Базовые понятия отладки и тестирования.	Перечень тем: 1.1.2. Отладка и тестирование. Методы отладки. Использование отладчика. Примеры построения тестовых сценариев. 3.9.2. Понятие исключительной ситуации. Обработка исключительных ситуаций.
Самостоятельная работа	
Тематика самостоятельной работы: Проработка конспекта, дополнительной литературы, поиск информации по заданным темам; подготовка к защите практических занятий, ответы на контрольные вопросы; подготовка к тестированию Выполнение тестирования и отладки программного продукта с протоколированием результатов.	
ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода.	
Уметь: У5. Оформлять код программы в соответствии со стандартом кодирования. У10. Проводить рефакторинг и оптимизацию программного кода.	Тематика лабораторных/практических занятий: ПЗ 36. Рефакторинг и оптимизация. Соответствие стандартам кодирования.
Знать: 35 - Стили и культура программирования. Стандарты кодирования. Рефакторинг и оптимизация кода.	Перечень тем: 3.10.1. Оптимизация кода. Примеры. Рефакторинг кода.
Самостоятельная работа	Тематика самостоятельной работы:

	Проработка конспекта, дополнительной литературы, поиск информации по заданным темам; подготовка к защите практических занятий, ответы на контрольные вопросы; подготовка к тестированию Выполнение индивидуальных заданий
ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.	
Уметь: У6. Выполнять тестирование, отладку кода программы. У8. Обрабатывать ошибки и исключительные ситуации;	Тематика лабораторных/практических занятий: ПЗ 15. Тестирование и отладка программы. ПЗ 35. Обработка исключений.
Знать: 311. Исключительные ситуации и ошибки 312. Базовые понятия отладки и тестирования.	Перечень тем: 3.1.1. Отладка и тестирование. Методы отладки. Использование отладчика. Примеры построения тестовых сценариев. 3.9.1. Понятие исключительной ситуации. Обработка исключительных ситуаций.
Самостоятельная работа	Тематика самостоятельной работы: Проработка конспекта, дополнительной литературы, поиск информации по заданным темам; подготовка к защите практических занятий, ответы на контрольные вопросы; подготовка к тестированию Выполнение тестирования и отладки программного продукта с протоколированием результатов.
ПК.2.5 Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.	
Уметь: У5. Оформлять код программы в соответствии со стандартом кодирования.	Тематика лабораторных/практических занятий: ПЗ 36. Рефакторинг и оптимизация. Соответствие стандартам кодирования.
Знать: 35. Стили и культура программирования. Стандарты кодирования. Рефакторинг и оптимизация кода.	Перечень тем: T.3.1.6. Стили программирования. Культура программного кода. Стандарты кодирования.
Самостоятельная работа	Тематика самостоятельной работы: Проработка конспекта, дополнительной литературы, поиск информации по заданным темам; подготовка к защите практических занятий, ответы на контрольные вопросы; подготовка к тестированию Выполнение индивидуальных заданий.

Технологии формирования ОК

Результаты (освоенные общие компетенции)	Основные показатели оценки результата
ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам.	<ul style="list-style-type: none"> - планирует деятельность по решению задачи в рамках заданных (известных) технологий, в том числе выделяя отдельные составляющие технологии; - анализирует потребности в ресурсах и планирует ресурсы в соответствии с заданным способом решения задачи
ОК 02. Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности.	<ul style="list-style-type: none"> - самостоятельно находит источник информации по заданному вопросу, пользуясь электронным или бумажным каталогом, справочно-библиографическими пособиями, поисковыми системами Интернета; - извлекает информацию по двум и более основаниям из одного или нескольких источников и систематизирует ее в рамках заданной структуры; - предлагает простую структуру для систематизации информации в соответствии с задачей информационного поиска; - делает вывод об объектах, процессах, явлениях на основе сравнительного анализа информации о них по заданным критериям или на основе заданных посылок и/или приводит аргументы в поддержку вывода - использует актуальные информационные технологии
ОК04. Эффективно взаимодействовать и работать в коллективе и команде	<ul style="list-style-type: none"> - взаимодействует с обучающимися, преподавателями, руководителями практик в ходе обучения.
ОК 5. Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста.	<ul style="list-style-type: none"> - коммуницирует на русском языке с обучающимися, преподавателями и прочими лицами устно и письменно - умеет четко формулировать мысли
ОК9. Пользоваться профессиональной документацией на государственном и иностранном языках.	<ul style="list-style-type: none"> - использует профессиональную документацию в процессе обучения и саморазвития

Лист изменений рабочей программы

Содержание изменения, страница рабочей программы	Дата и номер протокола заседания МК	Основание изменения