

**Министерство науки и высшего образования РФ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Национальный исследовательский университет «МЭИ»**

Направление подготовки/специальность: 13.03.02 Электроэнергетика и электротехника

Наименование образовательной программы: Электроэнергетика и электротехника

Уровень образования: высшее образование - бакалавриат

Форма обучения: Очная

**Оценочные материалы
по дисциплине
Программирование и основы алгоритмизации**

**Москва
2022**

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ РАЗРАБОТАЛ:

Преподаватель

(должность)

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Воробьева И.А.
	Идентификатор	R86e9a563-VorobyevaIA-80eec2d

(подпись)

И.А.

Воробьева

(расшифровка
подписи)

СОГЛАСОВАНО:

Руководитель
образовательной
программы

(должность, ученая степень, ученое
звание)

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Тулский В.Н.
	Идентификатор	R292b173d-TulskyVN-7e812984

(подпись)

В.Н.

Тулский

(расшифровка
подписи)

Заведующий
выпускающей кафедры

(должность, ученая степень, ученое
звание)

	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ»	
	Сведения о владельце ЦЭП МЭИ	
	Владелец	Тулский В.Н.
	Идентификатор	R292b173d-TulskyVN-7e812984

(подпись)

В.Н.

Тулский

(расшифровка
подписи)

ОБЩАЯ ЧАСТЬ

Оценочные материалы по дисциплине предназначены для оценки: достижения обучающимися запланированных результатов обучения по дисциплине, этапа формирования запланированных компетенций и уровня освоения дисциплины.

Оценочные материалы по дисциплине включают оценочные средства для проведения мероприятий текущего контроля успеваемости и промежуточной аттестации.

Формируемые у обучающегося компетенции:

1. УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач
ИД-1 Выполняет поиск необходимой информации, ее критический анализ и обобщает результаты анализа для решения поставленной задачи
2. ОПК-1 Способен понимать принципы работы современных информационных технологий и использования их для решения задач профессиональной деятельности
ИД-1 Применяет средства информационных, компьютерных и сетевых технологий для поиска, хранения, обработки, анализа и представления информации
3. ОПК-2 Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения
ИД-1 Алгоритмизирует решение задач и реализует алгоритмы с использованием программных средств

и включает:

для текущего контроля успеваемости:

Форма реализации: Компьютерное задание

1. (КР №1) Реализация основных алгоритмов, организация ввода и вывода в С и С++ (Контрольная работа)
2. (КР №2) Рекуррентные и рекурсивные алгоритмы (Контрольная работа)
3. (Тест №1) Области видимости, пространства имен, время жизни переменных (Тестирование)
4. (Тест №2) Динамические структуры данных (Тестирование)
5. (Тест №3) Базы данных. Инструменты многопоточного программирования (Тестирование)
6. Защита работы №1 "Бинарный поиск. Поиск подстроки в строке. Предложение модификаций" (Лабораторная работа)
7. Защита работы №10 "Методы приближения корней. Указатель на функцию в С(С++)" (Лабораторная работа)
8. Защита работы №2 "Реализация и анализ алгоритмов сортировки выбором и вставкой" (Лабораторная работа)
9. Защита работы №3 "Реализация стандарта NCR_R (русификация) на двух языках" (Лабораторная работа)
10. Защита работы №4 "Азбука Морзе. Разработка алгоритма кодирования и декодирования на двух языках" (Лабораторная работа)
11. Защита работы №5 "Реализация алгоритма пузырьковой сортировки в двух направлениях" (Лабораторная работа)
12. Защита работы №6 "Усовершенствованная пузырьковая сортировка. Экспериментальное исследование сортировок. Разработка и реализация экспериментальных тестов" (Лабораторная работа)

13. Защита работы №7 "Динамические массивы в качестве параметров подпрограмм в С(С++)" (Лабораторная работа)
14. Защита работы №8 "Динамические массивы и рекурсивные алгоритмы в С(С++)" (Лабораторная работа)
15. Защита работы №9 "Элементы функционального программирования в Python" (Лабораторная работа)
16. Защита расчетного задания №1 "Внешние библиотеки в Python. Графика в Python. Алгоритм имитации нормальных распределений объектов на плоскости с помощью равномерных распределений и их графическое отображение"" (Расчетно-графическая работа)
17. Защита расчетного задания №2 "Динамические структуры данных. Разработка собственного класса в Python. Запись и считывание объектов (работа с бинарными файлами). Реализация итерируемого объекта и генератора в классе" (Расчетно-графическая работа)

БРС дисциплины

2 семестр

Раздел дисциплины	Веса контрольных мероприятий, %									
	Индекс КМ:	КМ-1	КМ-2	КМ-3	КМ-4	КМ-5	КМ-6	КМ-7	КМ-8	КМ-9
	Срок КМ:	4	4	8	8	12	12	12	14	16
Основные аспекты реализаций Python и С(С++)										
Введение в язык С(С++). Сравнение реализаций Python и С(С++)	+				+					
Основы анализа алгоритмов										
Введение. Методы поиска, сортировки, кодирования и декодирования			+	+		+	+	+		
Исследование и проектирование алгоритмов										
Исследование алгоритмов. Внутренние библиотеки в Python									+	
Проектирование алгоритмов. Внешние библиотеки и графика в Python										+
Вес КМ:		5	5	5	10	5	15	5	25	25

3 семестр

Раздел дисциплины	Веса контрольных мероприятий, %									
	Индекс КМ:	КМ-10	КМ-11	КМ-12	КМ-13	КМ-14	КМ-15	КМ-16	КМ-17	
	Срок КМ:	4	4	8	8	12	14	15	16	
Взаимодействие программ с вызывающим контекстом										
Взаимодействие программ с вызывающим контекстом С(С++). Рекурсия	+	+	+			+				

Особенности динамической типизации. Способы поточной обработки данных								
Особенности динамической типизации. Способы поточной обработки данных				+				
Дополнительные возможности программирования в Python								
Структуры данных. Обзор методов построения и анализа алгоритмов						+	+	
Введение в объектно-ориентированное программирование (ООП) в Python								+
Вес КМ:	10	10	15	10	10	7	8	30

\$Общая часть/Для промежуточной аттестации\$

СОДЕРЖАНИЕ ОЦЕНОЧНЫХ СРЕДСТВ ТЕКУЩЕГО КОНТРОЛЯ

I. Оценочные средства для оценки запланированных результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций

Индекс компетенции	Индикатор	Запланированные результаты обучения по дисциплине	Контрольная точка
УК-1	ИД-1 _{УК-1} Выполняет поиск необходимой информации, ее критический анализ и обобщает результаты анализа для решения поставленной задачи	<p>Знать:</p> <p>аналитические и практические способы анализа сложности и эффективности алгоритмов численные и алгоритмические методы решения типовых задач поиска, сортировки, фильтрации, кодирования, приближенных решений математических задач</p> <p>Уметь:</p> <p>составлять алгоритмы решения задач по словесному описанию методов, производить сравнительный анализ алгоритмов по различным оценкам качества аналитически и на реальных тестах адаптировать, модифицировать и комбинировать</p>	<p>Защита работы №1 "Бинарный поиск. Поиск подстроки в строке. Предложение модификаций" (Лабораторная работа)</p> <p>Защита работы №2 "Реализация и анализ алгоритмов сортировки выбором и вставкой" (Лабораторная работа)</p> <p>Защита работы №4 "Азбука Морзе. Разработка алгоритма кодирования и декодирования на двух языках" (Лабораторная работа)</p> <p>Защита работы №5 "Реализация алгоритма пузырьковой сортировки в двух направлениях" (Лабораторная работа)</p> <p>Защита работы №6 "Усовершенствованная пузырьковая сортировка. Экспериментальное исследование сортировок. Разработка и реализация экспериментальных тестов" (Лабораторная работа)</p> <p>Защита работы №10 "Методы приближения корней. Указатель на функцию в C(C++)" (Лабораторная работа)</p>

		существующие алгоритмы для решения задач поиска, сортировки, фильтрации данных, кодирования и преобразования данных	
ОПК-1	ИД-1 _{ОПК-1} Применяет средства информационных, компьютерных и сетевых технологий для поиска, хранения, обработки, анализа и представления информации	<p>Знать:</p> <p>встроенные и внешние библиотеки языков основные принципы хранения, доступа и обработки данных в динамических структурах, базах данных, при поточном и многопоточном программировании</p> <p>Уметь:</p> <p>использовать возможности встроенных и внешних библиотек, а также графические библиотеки для визуализации решений</p> <p>программировать собственные классы для динамических объектов на основании принятых базовых операций в заданных структурах данных с применением их в прикладных задачах</p>	<p>Защита расчетного задания №1 "Внешние библиотеки в Python. Графика в Python. Алгоритм имитации нормальных распределений объектов на плоскости с помощью равномерных распределений и их графическое отображение"" (Расчетно-графическая работа) (Тест №2) Динамические структуры данных (Тестирование) (Тест №3) Базы данных. Инструменты многопоточного программирования (Тестирование)</p> <p>Защита расчетного задания №2 "Динамические структуры данных. Разработка собственного класса в Python. Запись и считывание объектов (работа с бинарными файлами). Реализация итерируемого объекта и генератора в классе" (Расчетно-графическая работа)</p>
ОПК-2	ИД-1 _{ОПК-2} Алгоритмизирует решение задач и реализует	<p>Знать:</p> <p>основные концепции программирования</p>	<p>(Тест №1) Области видимости, пространства имен, время жизни переменных (Тестирование)</p> <p>(КР №1) Реализация основных алгоритмов, организация ввода и</p>

	<p>алгоритмы использованием программных средств</p>	<p>с</p> <p>понятие рекуррентных соотношений и методы рекурсии принципы и особенности программирования в языках со статической и динамической типизацией Уметь: применять языковые возможности поточной обработки данных для типичных задач решать задачи с применением рекуррентного и рекурсивного подхода, использовать динамические структуры данных программировать решения типовых задач в языках с отличающейся семантикой</p>	<p>вывода в С и С++ (Контрольная работа) Защита работы №3 "Реализация стандарта NCR_R (русификация) на двух языках" (Лабораторная работа) Защита работы №7 "Динамические массивы в качестве параметров подпрограмм в С(С++)" (Лабораторная работа) Защита работы №8 "Динамические массивы и рекурсивные алгоритмы в С(С++)" (Лабораторная работа) (КР №2) Рекуррентные и рекурсивные алгоритмы (Контрольная работа) Защита работы №9 "Элементы функционального программирования в Python" (Лабораторная работа)</p>
--	---	---	---

II. Содержание оценочных средств. Шкала и критерии оценивания

2 семестр

КМ-1. (Тест №1) Области видимости, пространства имен, время жизни переменных

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Тестирование

Вес контрольного мероприятия в БРС: 5

Процедура проведения контрольного мероприятия: В текстовом редакторе через интернет. Расчетное время 15 минут, для вычислений можно использовать технические средства. Варианты заданий вычисляются по номеру студента в ведомости БАРС. Результаты сохраняются в ОСЭП МЭИ (либо в тестовой системе Cisco Webex)

Краткое содержание задания:

Работа ориентирована на проверку знаний по теме “Сравнение реализаций Python и C (C++)”

Контрольные вопросы/задания:

Знать: принципы и особенности программирования в языках со статической и динамической типизацией	<ol style="list-style-type: none">1.Как получить доступ к изменению глобальной переменной в Python из локальной области видимости? (1 балл)2.Можно ли прочитать содержимое глобальной переменной в Python из локальной области видимости? (1 балл)3.Каково назначение оператора nonlocal в Python? (2 балла)4.Укажите все способы подключения модулей в Python. (2 балла)5.Укажите способы подключения встроенных модулей и собственных модулей в C(C++). (2 балла)6.Как сделать именованное пространство имен в C++ ? (2 балла)7.Как связаны такие понятия, как “область видимости” и “время жизни” переменной? (3 балла)8.Как связаны такие понятия, как “область видимости переменной” и “пространство имен”? (3 балла)9.Как получить доступ к переменной, расположенной в глобальной области видимости, из локальной области в C (C++)? (2 балла)10.Какой модуль является головным модулем в C(C++)? В Python? (2 балла)
--	---

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 80

Описание характеристики выполнения знания: Оценка "отлично" выставляется если в результате ответов было набрано не менее 16-ти баллов

Оценка: 4

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если в результате ответов было набрано не менее 12-ти баллов

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если в результате ответов было набрано не менее 10-ти баллов

КМ-2. Защита работы №1 "Бинарный поиск. Поиск подстроки в строке.

Предложение модификаций"

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 5

Процедура проведения контрольного мероприятия: Заранее определяется общее задание на группу (с индивидуализацией по вариантам конкретной задачи) для самостоятельной работы в качестве домашнего задания. Необходимо реализовать программный код решения, сопроводить решение спецификацией-отчетом. Защита состоит в демонстрации работоспособности программы на тестах (или с помощью контрольной формулы) в компьютерных классах. Необходимо ответить на контрольные вопросы преподавателя по теме работы

Краткое содержание задания:

- 1.1. Реализовать алгоритм **бинарного поиска** на языке Python, изложенный в форме псевдокода в конспекте занятий. Провести серию тестов **с целью исследования поведения алгоритма** бинарного поиска для различных данных. Сделать выводы. Предложить модификацию для устранения выявленных проблем. Для **записи результата в файл** освоить код, данный в форме примера функций считывания и записи матриц из/в текстовый файл. Реализовать самостоятельно запись результатов бинарного поиска и исходных массивов в текстовый файл.
- 1.2. Реализовать **«наивный» алгоритм поиска подстрок в строке** на языке Python, изложенный в форме псевдокода в конспекте занятий. Провести серию тестов **с целью исследования поведения алгоритма и проверки его функциональности** на различных данных. Тесты придумать самостоятельно, сделать выводы

Контрольные вопросы/задания:

<p>Знать: численные и алгоритмические методы решения типовых задач поиска, сортировки, фильтрации, кодирования, приближенных решений математических задач</p>	<p>1. Задача 1.1. Поиск можно осуществлять на целых, вещественных числах или на символах. Если N – номер варианта, то: для $N \bmod 2 = 0$ – работа с числами; для $N \bmod 2 = 1$ – работа с символами</p> <p>Контрольный вопрос. Можно ли осуществить поиск не на одиночных символах, а на элементах-“строках”? По какому правилу будет осуществляться сравнение таких элементов-“строк” на “больше”, “меньше”, “равно”?</p> <p>2. Задача 1.2. «Наивный алгоритм поиска подстрок в строке» не предусматривает признак неуспеха поиска — предложить модификацию для устранения этого недостатка. Алгоритм ищет все подстроки в строке — предложить модификацию,</p>
---	--

	<p>которая будет останавливаться при первом же найденном совпадении и содержит признак неуспеха поиска</p> <p>Контрольный вопрос. Как влияет выход по команде break из параметрического (итерационного) цикла в конструкции for...in...else (while...else) в языке Python на порядок выполнения операторов? В каком случае будет отработана ветвь "else"?</p>
--	--

Описание шкалы оценивания:

Оценка: зачтено

Описание характеристики выполнения знания: Оценка "зачтено" выставляется если задание выполнено правильно или с незначительными недочетами

Оценка: не зачтено

Описание характеристики выполнения знания: Оценка "не зачтено" выставляется если задание не выполнено в отведенный срок или результат не соответствует заданию

КМ-3. Защита работы №2 "Реализация и анализ алгоритмов сортировки выбором и вставкой"

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 5

Процедура проведения контрольного мероприятия: Заранее определяется общее задание на группу (с индивидуализацией по вариантам конкретной задачи) для самостоятельной работы в качестве домашнего задания. Необходимо реализовать программный код решения, сопроводить решение спецификацией-отчетом. Защита состоит в демонстрации работоспособности программы на тестах (или с помощью контрольной формулы) в компьютерных классах. Необходимо ответить на контрольные вопросы преподавателя по теме работы

Краткое содержание задания:

1.1. Реализовать алгоритм **сортировка вставкой** на языке Python, изложенный в форме псевдокода в конспекте занятий.

1.2. Реализовать алгоритм **сортировка выбором** на языке Python, изложенный в форме псевдокода в конспекте занятий.

Сортировку можно осуществлять на целых, вещественных числах или на символах. Если **N** – номер варианта, то: для $N \bmod 3=0$ – работа с целыми числами; для $N \bmod 3=1$ – работа с вещественными числами; для $N \bmod 3=2$ – работа с символами.

Провести серию тестов **с целью исследования правильного поведения алгоритмов** для различных исходных данных. Чтобы проверить работу сортировки необходимо выводить промежуточное состояние массива на каждом шаге сортировки. Например, если массив длины **t**, то выводов состояния массивов будет **t-1**

Контрольные вопросы/задания:

<p>Знать: численные и алгоритмические методы решения типовых задач поиска, сортировки, фильтрации, кодирования, приближенных решений математических задач</p>	<p>1.Обладает ли сортировка вставкой свойством устойчивости?</p> <p>2.Обладает ли сортировка выбором свойством естественности?</p> <p>3.Какова оценка сложности сверху для алгоритмов сортировки вставкой (выбором)?</p>
---	--

<p>Уметь: адаптировать, модифицировать и комбинировать существующие алгоритмы для решения задач поиска, сортировки, фильтрации данных, кодирования и преобразования данных</p>	<p>1. Как сэкономить на количестве обменов в сортировке вставкой? 2. Пусть имеется алгоритм сортировки выбором, который работает от “головы” массива и упорядочивает его элементы “по возрастанию” значений. Предложите два способа изменения этого алгоритма, чтобы упорядочивание происходило “по убыванию” значений 3. Предложите такую модификацию вашего алгоритма сортировки вставкой, чтобы результат упорядочивания значений элементов поменялся на противоположный</p>
--	---

Описание шкалы оценивания:

Оценка: зачтено

Описание характеристики выполнения знания: Оценка "зачтено" выставляется если задание выполнено правильно или с незначительными недочетами

Оценка: не зачтено

Описание характеристики выполнения знания: Оценка "не зачтено" выставляется если задание не выполнено в отведенный срок или результат не соответствует заданию

КМ-4. (КР №1) Реализация основных алгоритмов, организация ввода и вывода в С и С++

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Контрольная работа

Вес контрольного мероприятия в БРС: 10

Процедура проведения контрольного мероприятия: Проводится на лабораторном (или практическом) занятии в форме реализации рабочей программы в любой среде, позволяющей выполнять отладку и запуск на языке программирования С (С++), что обеспечивает возможность самоконтроля при выполнении решения. На выполнение задания отводится 60 минут, работа отправляется в текстовом формате по почте ОСЭП

Краткое содержание задания:

Массивы определяются как встроенные (неизменяемого размера) массивы языка соответствующего типа данных. Необходимо заполнить массив данными с клавиатуры (поток ввода) и вывести на экран (поток вывода). Также вывести результат работы программы в форматированном виде. Ввод и вывод можно осуществлять, как в С стиле, так и в С++ стиле. Соблюдать форматирование кода, сопроводить код комментированием, сопроводить текст кода заданием, примером “строки ввода” данных и указанием ожидаемого результата (полное представление)

Контрольные вопросы/задания:

<p>Знать: принципы и особенности программирования в языках со статической и динамической типизацией</p>	<p>1. Дан массив A из f вещественных чисел. Определить наибольшую по абсолютной величине разность между A_i и A_{i-1} 2. Дан массив A длины g из вещественных чисел. Определить модуль разности наименьшего и наибольшего элементов в массиве A</p>
---	---

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 85

Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

Оценка: 4

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто. Выбрано верное направление для решения задачи, однако код содержит незначительные синтаксические ошибки.

Оценка: 3

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

КМ-5. Защита работы №3 "Реализация стандарта NCR_R (русификация) на двух языках"

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 5

Процедура проведения контрольного мероприятия: Заранее определяется общее задание на группу (группа разбивается на бригады по 2-3 человека) для самостоятельной и коллективной работы в качестве домашнего задания. Необходимо реализовать программный код решения, сопроводить решение спецификацией-отчетом (один на бригаду). Защита состоит в демонстрации работоспособности программы на тестах (или с помощью контрольной формулы) в компьютерных классах. Необходимо ответить на контрольные вопросы преподавателя по теме работы. На защите каждый член бригады отвечает за определенную часть выполненной работы полностью

Краткое содержание задания:

Реализовать алгоритм для решения задачи "русификация по стандарту NCR_R" в виде функции на языке C(C++), а затем модифицировать идею для реализации аналогичной функции на языке Python. Сущности и конструкции языка Python студенты определяют самостоятельно, по своему выбору. Отчеты должны содержать: задание, код, проверочные тесты. Желательно подумать над сравнением реализаций в C и Python с точки зрения удобства использования языковых конструкций.

Уточнение задачи. Исходная строка декодирует символы строчного латинского алфавита в заглавные символы кириллицы, расположенные на соответствующих клавишах клавиатуры, например строку вида "vjcrdf" в строку вида "МОСКВА". Отдельно стоят символы по нажатию клавиши Shift:

~ = Ё, { = X, } = Ъ, | = Ж, < = Б, > = Ю и без Shift ' = Э.

Все остальные символы исходного сообщения остаются без изменений. Например, строка 'HELLO, ~|br!' конвертируется в строку 'HELLO, ЁЖИК!'

Пример реализации алгоритма на языке C(C++) рассматривается на практических занятиях

Контрольные вопросы/задания:

Уметь: программировать решения типовых задач в языках	1.Напишите функцию на C(C++), которая приведет все символы латинского алфавита к заглавным, а
---	---

с отличающейся семантикой	<p>остальные оставит без изменений</p> <p>2. Напишите функцию на Python, которая изменит регистр всех символов латинского алфавита, а остальные оставит без изменений</p> <p>3. Напишите функцию на C(C++), которая приведет символы кириллицы из кодировки CP866 в кодировку CP1251. С какими трудностями можно столкнуться при решении подобной задачи, если программу перенести в те компиляторы языка C++, которые используют юникод?</p>
---------------------------	---

Описание шкалы оценивания:

Оценка: зачтено

Описание характеристики выполнения знания: Оценка "зачтено" выставляется каждому члену бригады индивидуально, если задание выполнено правильно или с незначительными недочетами, даны ответы на вопросы преподавателя

Оценка: не зачтено

Описание характеристики выполнения знания: Оценка "не зачтено" выставляется если задание не выполнено в отведенный срок или результат не соответствует заданию, либо на защите студент не смог ответить на вопросы преподавателя по своей индивидуальной части работы

КМ-6. Защита работы №4 "Азбука Морзе. Разработка алгоритма кодирования и декодирования на двух языках"

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 15

Процедура проведения контрольного мероприятия: Заранее определяется общее задание на группу (группа разбивается на бригады по 2-3 человека) для самостоятельной и коллективной работы в качестве домашнего задания. Необходимо реализовать программный код решения, сопроводить решение спецификацией-отчетом (один на бригаду). Защита состоит в демонстрации работоспособности программы на тестах (или с помощью контрольной формулы) в компьютерных классах. Необходимо ответить на контрольные вопросы преподавателя по теме работы. На защите каждый член бригады отвечает за определенную часть выполненной работы полностью

Краткое содержание задания:

Придумать алгоритмы для решения задач "кодирования и декодирования азбуки Морзе" и реализовать их в виде функций на языках C(C++) и Python. Сущности и конструкции языков для реализации алгоритмов, студенты определяют самостоятельно, по своему выбору. Отчеты должны содержать: задание, код, проверочные тесты. Желательно подумать над сравнением реализаций в C и Python с точки зрения удобства использования языковых конструкций.

Уточнение задачи. Дана таблица кодирования латинского заглавных букв и цифр в "буквы" азбуки Морзе. Для обеспечения совместимости между вариантами, договоримся использовать короткие сообщения, помещающиеся в 255 символов. Хранить сообщения можно в обычном текстовом файле. Считывать данные можно как из текстового файла (в Python), так и обычным вводом строки (C, C++).

Поясним разделения между символами, словами и предложениями. Скорость передачи при физической реализации азбуки Морзе считается относительно длительности звучания точки. Длительность всех остальных элементов считают по схеме:

$$"- " = 3 * "."$$

$$\text{пауза между буквами} = 1 * "."$$

$$\text{пауза между словами} = 3 * "."$$

$$\text{пауза между предложениями} = 7 * "."$$

Паузу, равную одной точке, будем кодировать символом пробела. Схема немного отличается от реальной «звуковой», так как в передаче сигнала не по звуку, а в текстовом виде нет необходимости отделять “точку” от “тире”

Контрольные вопросы/задания:

Уметь: адаптировать, модифицировать и комбинировать существующие алгоритмы для решения задач поиска, сортировки, фильтрации данных, кодирования и преобразования данных	1.Что следует изменить в функции, написанной под кодировку символов, в которой они занимают ровно один байт, чтобы функция могла работать с символами кодируемыми двумя (четырьмя) байтами? 2.Покажите, как можно учесть в исходном “кодируемом” тексте наличие знаков препинания? 3.Покажите, где в ваших функциях полезно использовать словари (ассоциированные массивы)? Как?
---	---

Описание шкалы оценивания:

Оценка: зачтено

Описание характеристики выполнения знания: Оценка "зачтено" выставляется каждому члену бригады индивидуально, если задание выполнено правильно или с незначительными недочетами, даны ответы на вопросы преподавателя

Оценка: не зачтено

Описание характеристики выполнения знания: Оценка "не зачтено" выставляется если задание не выполнено в отведенный срок или результат не соответствует заданию, либо на защите студент не смог ответить на вопросы преподавателя по своей индивидуальной части работы

КМ-7. Защита работы №5 "Реализация алгоритма пузырьковой сортировки в двух направлениях"

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 5

Процедура проведения контрольного мероприятия: Заранее определяется общее задание на группу (с индивидуализацией по вариантам конкретной задачи) для самостоятельной работы в качестве домашнего задания. Необходимо реализовать программный код решения,

сопроводить решение спецификацией-отчетом. Защита состоит в демонстрации работоспособности программы на тестах (или с помощью контрольной формулы) в компьютерных классах. Необходимо ответить на контрольные вопросы преподавателя по теме работы

Краткое содержание задания:

Реализовать алгоритм **пузырьковая сортировка** на языке Python, изложенный в форме псевдокода в конспекте занятий.

Сортировку можно осуществлять на целых, вещественных числах или на символах. Если N – номер варианта, то: для $N \bmod 3=0$ – работа с вещественными числами; для $N \bmod 3=1$ – работа с символами; для $N \bmod 3=2$ – работа с целыми числами.

Сортировку необходимо реализовать в ДВУХ направлениях. Если N – номер варианта, то: а) для $N \bmod 4=0$ – сортировка по возрастанию (неубыванию) ($\max \rightarrow$ в конец ***I*** $\min \rightarrow$ в начало); б) для $N \bmod 4=1$ – сортировка по убыванию (невозрастанию) ($\min \rightarrow$ в конец ***I*** $\max \rightarrow$ в начало); в*) для $N \bmod 4=2$ – сортировка по возрастанию (неубыванию) ($\max \rightarrow$ в конец ***I*** $\min \rightarrow$ в начало); г*) для $N \bmod 4=3$ – сортировка по убыванию (невозрастанию) ($\min \rightarrow$ в конец ***I*** $\max \rightarrow$ в начало)

**) для вариантов, помеченных звездочкой, накладывается дополнительное условие: сортировка числовых величин осуществляется по их абсолютной величине, либо, если сортируются символы, то идентичными считаются символы в верхнем и нижнем регистрах, например считается, что 'z' = 'Z'*

Провести серию тестов с целью исследования правильного поведения алгоритмов для различных исходных данных. Чтобы проверить работу сортировки необходимо выводить промежуточное состояние массива на каждом шаге сортировки. Например, если массив длины t , то выводов состояния массивов будет $t-1$.

Контрольные вопросы/задания:

<p>Уметь: адаптировать, модифицировать и комбинировать существующие алгоритмы для решения задач поиска, сортировки, фильтрации данных, кодирования и преобразования данных</p>	<ol style="list-style-type: none"> 1. Как можно усовершенствовать стандартную пузырьковую сортировку (с целью сокращения шагов сортировки) в том случае, если массив оказался отсортирован досрочно? 2. Покажите в коде, что именно в алгоритме указывает, на “направление” выбранной сортировки? 3. Предложите решение следующей задачи, основанное на алгоритме пузырьковой сортировки: “массив, состоящий из символов ”цифр” и символов “латинского алфавита” отсортировать так, чтобы сначала в нем располагались все “цифры”, а затем символы “латинского алфавита”
--	---

Описание шкалы оценивания:

Оценка: зачтено

Описание характеристики выполнения знания: Оценка "зачтено" выставляется если задание выполнено правильно или с незначительными недочетами

Оценка: не зачтено

Описание характеристики выполнения знания: Оценка "не зачтено" выставляется если задание не выполнено в отведенный срок или результат не соответствует заданию

КМ-8. Защита работы №6 "Усовершенствованная пузырьковая сортировка. Экспериментальное исследование сортировок. Разработка и реализация экспериментальных тестов"

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 25

Процедура проведения контрольного мероприятия: Заранее определяется общее задание на группу (с индивидуализацией по вариантам конкретной задачи) для самостоятельной работы в качестве домашнего задания. Необходимо реализовать программный код решения, сопроводить решение спецификацией-отчетом. Защита состоит в демонстрации работоспособности программы на тестах (или с помощью контрольной формулы) в компьютерных классах. Необходимо ответить на контрольные вопросы преподавателя по теме работы

Краткое содержание задания:

Реализовать два алгоритма пузырьковых сортировок: стандартную сортировку (назовем ее *Bubble*) и усовершенствованную (метод флажка с досрочным выходом, назовем ее *Bubble_F*). Все необходимые алгоритмы в программе реализуются в виде функций. Проводится серия сравнительных экспериментов над алгоритмами. Результаты экспериментов сводятся в соответствующие таблицы. Данные таблиц анализируются, на основании анализа делают выводы, формируется спецификация-отчет. Выбирается сортировка в **любом направлении (голова-хвост)** по желанию. Если N – номер варианта, то: для $N \bmod 3=0$ – работа с вещественными числами; для $N \bmod 3=1$ – работа с символами; для $N \bmod 3=2$ – работа с целыми числами. Дополнительно: для четных вариантов — сортировка по неубыванию ($\max \rightarrow$ в конец **ИЛИ** $\min \rightarrow$ в начало); для нечетных вариантов — сортировка по невозрастанию ($\min \rightarrow$ в конец **ИЛИ** $\max \rightarrow$ в начало)

Контрольные вопросы/задания:

Знать: аналитические и практические способы анализа сложности и эффективности алгоритмов	<ol style="list-style-type: none">1. Чем отличаются методы <i>time()</i> и <i>ctime()</i> из модуля <i>time</i>?2. Какой метод из модуля <i>time</i> останавливает ход выполнения программы ровно на t секунд (доли секунд)?3. Что представляет собой генератор псевдо-случайных чисел?4. Для чего необходим метод <i>random.seed([t])</i>?
--	---

Описание шкалы оценивания:

Оценка: зачтено

Описание характеристики выполнения знания: Оценка "зачтено" выставляется если задание выполнено правильно или с незначительными недочетами

Оценка: не зачтено

Описание характеристики выполнения знания: Оценка "не зачтено" выставляется если задание не выполнено в отведенный срок или результат не соответствует заданию

КМ-9. Защита расчетного задания №1 "Внешние библиотеки в Python. Графика в Python. Алгоритм имитации нормальных распределений объектов на плоскости с помощью равномерных распределений и их графическое отображение"

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Расчетно-графическая работа

Вес контрольного мероприятия в БРС: 25

Процедура проведения контрольного мероприятия: Заранее определяется общее задание на группу для самостоятельной работы в качестве домашнего задания. Необходимо реализовать программный код решения, сопроводить решение спецификацией-отчетом. Защита состоит в демонстрации работоспособности программы на тестах (или с помощью контрольной формулы) в компьютерных классах. Необходимо ответить на контрольные вопросы преподавателя по теме работы

Краткое содержание задания:

Задание. Освоить подключение внешней графической библиотеки в языке Python, освоить работу с графическими примитивами в библиотеке, реализовать моделирование узлов некоторой сети с помощью предложенного алгоритма *XY_MND* и визуализировать результат работы алгоритма средствами графической библиотеки. Требуется смоделировать не менее четырех различных сетей с различными входными параметрами для алгоритма *XY_MND*.

Отчет включает следующую информацию:

- а) задание;
- б) описание (использованный способ) подключения графической библиотеки (например, *graphics*);
- в) описание не менее 7 основных методов, предоставляемых графической библиотекой, с фрагментом кода и иллюстрацией результата его работы;
- г) код основной программы, содержащий функции моделирования сети и функции отображения результата моделирования;
- д) иллюстрации результатов моделирования сетей (распределение точек на плоскости) с указанием всех используемых входных параметров алгоритма моделирования сети, для которой представлена иллюстрация

Контрольные вопросы/задания:

Знать: встроенные и внешние библиотеки языков	1. На какой теореме базируется предложенный алгоритм <i>XY_MND</i> ? 2. Что утверждает "правило двух(трех) сигм"? Какое правило более строгое? 3. Какие основные характеристики есть у таких графических примитивов, как "фигура на плоскости"?
Уметь: использовать возможности встроенных и внешних библиотек, а также графические библиотеки для визуализации решений	1. Укажите способ перевода физической величины (точки на вещественной оси) в точку отображения на графическом полотне 2. Предложите способы реализации фигуры, не входящей в реализованные в библиотеке графические примитивы, например, "улыбку" или "крыло бабочки". 3. Укажите, как вы реализуете вывод на плоскости график какой-либо функции от одного аргумента, причем так, чтобы были показаны все четверти плоскости.

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 85

Описание характеристики выполнения знания: Оценка "отлично" выставляется если задание выполнено в полном объеме или выполнено преимущественно верно

Оценка: 4

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если большинство вопросов раскрыто, выбрано верное направление для решения задач

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если задание преимущественно выполнено

3 семестр

КМ-10. Защита работы №7 "Динамические массивы в качестве параметров подпрограмм в С(С++)"

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 10

Процедура проведения контрольного мероприятия: Заранее определяется общее задание на группу (с индивидуализацией по вариантам конкретной задачи) для самостоятельной работы в качестве домашнего задания. Необходимо реализовать программный код решения, сопроводить решение спецификацией-отчетом. Защита состоит в демонстрации работоспособности программы на тестах (или с помощью контрольной формулы) в компьютерных классах. Необходимо ответить на контрольные вопросы преподавателя по теме работы

Краткое содержание задания:

1. В задачах заданы одномерные массивы (вектор, вектор-столбец) и двумерные массивы (матрицы). Размерность массивов определяется смыслом задачи.
2. Написать программу решения задачи, используя язык программирования С (С++) (можно использовать любую доступную среду разработки).
3. При решении задачи использовать динамическое распределение памяти.
4. Использовать подпрограммы, реализующие функции: выделение памяти под массив (и/или матрицу), освобождение, выделенной памяти, ввод значений в массив (и/или матрицу), вывод массива (матрицы) на консоль.
5. Для непосредственного решения задачи, выделить не менее одной вспомогательной функции, помимо функций из п. 4.
6. Осуществить проверку на аномалии для размерностей массивов в соответствии с задачей, например при умножении матриц, необходимо проверить их согласованность (если не сказано иное, векторы считаются заданными в n -мерном евклидовом пространстве).
7. Провести серию тестов с целью проверки работоспособности кода, а также рассматривая вырожденные случаи размерностей массивов

Контрольные вопросы/задания:

Уметь: решать задачи с применением рекуррентного и	1.«Транспонировать» матрицу A относительно побочной диагонали, а матрицу B — относительно
--	---

рекурсивного использовать структуры данных	подхода, динамические	главной 2.Сдвинуть матрицу <i>A</i> циклически относительно столбцов: 1-й столбец на место 2-го, 2-й на место 3-го и т.д. 3.Вычислить определители третьего порядка для заданных матриц <i>A</i> и <i>B</i>
--	--------------------------	---

Описание шкалы оценивания:

Оценка: зачтено

Описание характеристики выполнения знания: Оценка "зачтено" выставляется если задание выполнено правильно или с незначительными недочетами

Оценка: не зачтено

Описание характеристики выполнения знания: Оценка "не зачтено" выставляется если задание не выполнено в отведенный срок или результат не соответствует заданию

КМ-11. Защита работы №8 "Динамические массивы и рекурсивные алгоритмы в C(C++)"

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 10

Процедура проведения контрольного мероприятия: Заранее определяется общее задание на группу (с индивидуализацией по вариантам конкретной задачи) для самостоятельной работы в качестве домашнего задания. Необходимо реализовать программный код решения, сопроводить решение спецификацией-отчетом. Защита состоит в демонстрации работоспособности программы на тестах (или с помощью контрольной формулы) в компьютерных классах. Необходимо ответить на контрольные вопросы преподавателя по теме работы

Краткое содержание задания:

1. В задачах задан одномерный динамический массив длины *n*.
2. Написать программу решения задачи, используя язык программирования C (C++) (можно использовать любую доступную среду разработки).
3. Осуществить проверку на аномалию для размерности массива ($n \leq 0$).
4. Провести серию тестов с целью проверки работоспособности кода, а также рассматривая вырожденные случаи размерностей массивов. Рекомендации по размерностям: $n = 0, 1, 2$ (минимальная четная размерность), $n = 3$ (минимальное деление на нечетной размерности), $n = 9$ (нечетная), $n = 10$ (четная). Тестирование, связанное со значениями массива необходимо проводить согласно смыслу задачи по варианту.
5. **Оформить отчет:**
 - a. формулировка задачи с именованием данных согласно программе;
 - b. код программы с комментариями;
 - c. результаты тестирования, сведенные в таблицу;
 - d. нарисовать полное дерево рекурсивных вызовов для размерности $n = 9$, подсчитав в нем объем и глубину рекурсии

Контрольные вопросы/задания:

Уметь: решать задачи с применением рекуррентного и рекурсивного подхода,	1.Если потребуется писать рекурсивный алгоритм в условиях ограничения на размер памяти стека, какую рекурсию следует выбрать?
--	---

использовать динамические структуры данных	<p>2. Если потребуется писать рекурсивный алгоритм в условиях ограничения на время исполнения программы, какую рекурсию следует выбрать?</p> <p>Для заданного одномерного массива Y из N элементов с помощью рекурсивной функции проверить, что для всех элементов массива выполняется условие $Y_i < 0$. В рекурсивной функции каждый раз делить рассматриваемую часть массива пополам, проверяя выполнение условия с помощью этой же функции сначала в левой половине, а затем при необходимости и в правой половине. Рекурсивные вызовы заканчивать, когда останется только один элемент в рассматриваемой части массива.</p> <p>3. Например, для $N = 4$: $\int_{i=1}^4 (Y_i < 0) = \int_{i=1}^2 (Y_i < 0) \& \int_{i=3}^4 (Y_i < 0)$.</p> <p>Для заданного одномерного массива A из N элементов с помощью рекурсивной функции найти значение минимального элемента массива. В рекурсивной функции каждый раз делить рассматриваемую часть массива на две части: одну треть и две третьих, вычисляя минимум с помощью этой же функции сначала в левой части, а затем и в правой части. Рекурсивные вызовы заканчивать, когда останется только один элемент в рассматриваемой части массива.</p> <p>4. Например, для $N = 6$: $\min_{i=1}^6 A = \min \left(\min_{i=1}^2 A; \min_{i=3}^6 A \right)$.</p>
--	--

Описание шкалы оценивания:

Оценка: зачтено

Описание характеристики выполнения знания: Оценка "зачтено" выставляется если задание выполнено правильно или с незначительными недочетами

Оценка: не зачтено

Описание характеристики выполнения знания: Оценка "не зачтено" выставляется если задание не выполнено в отведенный срок или результат не соответствует заданию

КМ-12. (КР №2) Рекуррентные и рекурсивные алгоритмы

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Контрольная работа

Вес контрольного мероприятия в БРС: 15

Процедура проведения контрольного мероприятия: Проводится на лабораторном (или практическом) занятии в форме реализации рабочей программы в любой среде, позволяющей выполнять отладку и запуск на языке программирования C (C++), что обеспечивает возможность самоконтроля при выполнении решения. На выполнение задания отводится 80 минут, работа отправляется в текстовом формате по почте ОСЭП

Краткое содержание задания:

Задание. Даны задачи, решаемые с помощью циклических алгоритмов. Задачи имеют две формулировки: простую «А» — с постоянным воздействующим параметром; усложненную «Б» — модификацию «А» с изменяющимся воздействующим параметром.

Допустимые решения:

- с выводом рекуррентного соотношения, в цикле, с выводом на экран результатов работы каждого шага;
- использование рекурсивной функции для получения конечного результата за N шагов;
- использование рекурсивной функции с выводом на экран результатов работы каждого шага рекурсии

Контрольные вопросы/задания:

Знать: понятие рекуррентных соотношений и методы рекурсии	<p>1.</p> <table border="1"> <tr> <td data-bbox="735 1928 783 2072">А</td> <td data-bbox="783 1928 1495 2072">Зарботная плата служащего в январе составила Z руб. Вывести прогнозируемую зарплату для каждого месяца текущего года в предположении ежемесячного увеличения зарплаты на p % (по</td> </tr> </table>	А	Зарботная плата служащего в январе составила Z руб. Вывести прогнозируемую зарплату для каждого месяца текущего года в предположении ежемесячного увеличения зарплаты на p % (по
А	Зарботная плата служащего в январе составила Z руб. Вывести прогнозируемую зарплату для каждого месяца текущего года в предположении ежемесячного увеличения зарплаты на p % (по		

		от-ношению к предыдущему месяцу)
	Б	<i>В феврале предполагаемый коэффициент индексации заработной платы равен p %, а в каждом следующем месяце уменьшается в 2 раза (по отношению к предыдущему месяцу)</i>
	2.	
	А	Открыт счет, содержащий Z руб. В начале каждого следующего месяца на него переводят Y руб. Хранимая сумма ежемесячно нарастает на p % (процентная ставка). Определить ее значение в конце 1, 2, ..., N -го месяца накопления
	Б	<i>В первом месяце ежемесячная процентная ставка равна p %, а в каждый следующий месяц увеличивается на 0,1 % (по отношению к предыдущему месяцу)</i>

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 80

Описание характеристики выполнения знания: Оценка "отлично" выставляется если правильно выполнено задание "Б" всеми способами "а) - с)"

Оценка: 4

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если правильно выполнено задание "А" способом "с)" И правильно выполнено решение "Б" способом "а)"

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если правильно выполнено задание "А" двумя способами "а) и с)" ИЛИ правильно выполнено решение "Б" способом "а)"

КМ-13. Защита работы №9 "Элементы функционального программирования в Python"

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 10

Процедура проведения контрольного мероприятия: Заранее определяется общее задание на группу (с индивидуализацией по вариантам конкретной задачи) для самостоятельной работы в качестве домашнего задания. Необходимо реализовать программный код решения, сопроводить решение спецификацией-отчетом. Защита состоит в демонстрации работоспособности программы на тестах (или с помощью контрольной формулы) в компьютерных классах. Необходимо ответить на контрольные вопросы преподавателя по теме работы

Краткое содержание задания:

Лабораторная работа предназначена для закрепления материала, изученного на практическом занятии: функции *map()*, *reduce()*, *filter()* и *zip()*; генераторы списков. Необходимо самостоятельно проработать вопрос, — в каких задачах могут быть полезны функции обработки данных в последовательностях и генераторы списков. Предлагается возможность выбора: а) применить новые возможности для задач, решаемых ранее (адреса расположения: [задачи 1 семестра](#); [задачи 2 семестра](#)); б) самостоятельно придумать задачу, сформулировать и решить.

При решении варианта б), стоит обратить внимание, что наиболее интересны задачи, связанные с обработкой различных данных составного типа, например данные о цветах в цветочном магазине (Название, Страна производитель, Дата поставки, Количество, Стоимость за единицу) и т.п. Также интересны задачи, связанные с обработкой текста, например «*текст, каждая строка которого содержит фамилию, упорядочить по алфавиту*» или «*распечатать все числа, встречающиеся в заданном тексте*».

Оформить спецификацию-отчет

Контрольные вопросы/задания:

<p>Знать: основные концепции программирования</p>	<p>1. Может ли функция <code>map()</code> работать с несколькими списками? 2. Что будет, если функция <code>map()</code> или <code>zip()</code> получат в качестве аргументов списки разной длины? 3. Можно ли при формировании списковой сборки использовать конструкцию "if"? Конструкцию "if...else"?</p>
<p>Уметь: применять языковые возможности поточной обработки данных для типичных задач</p>	<p>1. Пусть текстовый файл состоит из строк вида: 6 -12 3 12 -30 12 0 2 3 10 13 -7 6</p> <p>Сформируйте список, каждый элемент которого будет представлять собой список из пары чисел (расположенной в одной строке исходного файла) используя синтаксическую конструкцию "списковой сборки"</p> <p>2. Пусть задан список, содержащий целые числа. Создайте с помощью конструкции "списковой сборки" новый список, который будет состоять из квадратов элементов исходного списка</p> <p>3. Пусть задана строка произвольных символов. Создайте с помощью конструкции "списковой сборки" список, который будет состоять только из символов, изображающих цифры в исходной строке</p>

Описание шкалы оценивания:

Оценка: зачтено

Описание характеристики выполнения знания: Оценка "зачтено" выставляется если задание выполнено правильно или с незначительными недочетами

Оценка: не зачтено

Описание характеристики выполнения знания: Оценка "не зачтено" выставляется если задание не выполнено в отведенный срок или результат не соответствует заданию

КМ-14. Защита работы №10 "Методы приближения корней. Указатель на функцию в C(C++)"

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Лабораторная работа

Вес контрольного мероприятия в БРС: 10

Процедура проведения контрольного мероприятия: Заранее определяется общее задание на группу (с индивидуализацией по вариантам конкретной задачи) для самостоятельной работы в качестве домашнего задания. Необходимо реализовать программный код решения, сопроводить решение спецификацией-отчетом. Защита состоит в демонстрации работоспособности программы на тестах (или с помощью контрольной формулы) в компьютерных классах. Необходимо ответить на контрольные вопросы преподавателя по теме работы

Краткое содержание задания:

1. Целью работы является изучение приближенных методов по их словесному описанию и самостоятельное написание алгоритма, а также освоение способа передачи функции в качестве аргумента другой функции.
2. Методы описаны в отдельно прилагаемом файле «Методы нахождения корня функционального уравнения.pdf»[\[1\]](#).
3. В качестве примера выбрана задача нахождения корня функционального уравнения на отрезке **[A; B]**. Задача, сформулированная для базового метода **бисекции**[\[2\]](#), и варианты уравнений приведены в файле «Варианты задач вычисления корня уравнения.pdf».
4. Необходимо составить алгоритм и написать программу решения задачи, используя язык программирования C (C++), двумя методами. Первый метод — это метод бисекции, а второй:
 - a. метод Ньютона(касательных) для вариантов $3*k+1$;
 - b. метод секущих для вариантов $3*k+2$;
 - c. метод хорд для вариантов $3*k+3$, где $k=0,1,2,\dots$
5. Каждый метод должен быть решен для двух функций: функция **N** и функция **N+5** из списка вариантов (п.3), где **N** – номер по журналу. Должен быть задействован способ передачи функции в реализуемый метод с помощью указателя на функцию.
6. Осуществить проверку на аномалию для заданного отрезка: $0 < A < B < 2$.
7. Провести серию тестов для двух методов и двух функций в каждом из методов и точности **Eps = 10^{-2} , 10^{-4} , 10^{-6} , 10^{-8}** . На экран должна выводиться следующая информация:
 - a. метод и номер функции (один раз на серию тестов);
 - b. точность;
 - c. решение уравнения;
 - d. количество итераций для заданной точности;
 - e. значение левой части уравнения, если в качестве аргумента подставить найденный корень уравнения.
8. **Оформить отчет:**
 - a. формулировка задачи с именованием данных согласно программе;
 - b. краткое изложение метода, кроме метода деления отрезка пополам;
 - c. код программы с комментариями;
 - d. результаты тестирования, сведенные в таблицу

[1] Гречкина, П.В. Программирование, ч. 1, Электронный учебно-методический комплекс по направлению 230100 Информатика и вычислительная техника / П.В. Гречкина. – Москва, 2012 г. © НИУ МЭИ (или Глаголев В.Б. Visual Basic 6.0. Сборник заданий для лабораторно- практических занятий. Методическое пособие по курсу Информатика... МЭИ,2000).

[2] Он же — метод деления отрезка пополам (метод дихотомии, метод проб).

Контрольные вопросы/задания:

<p>Уметь: составлять алгоритмы решения задач по словесному описанию методов, производить сравнительный анализ алгоритмов по различным оценкам качества аналитически и на реальных тестах</p>	<ol style="list-style-type: none"> 1. Пусть есть функции вида <code>int** foo(char*, double);</code> и <code>void bar();</code> Как синтаксически описать формальные аргументы функции <code>int** baz(____, ____)</code> так, чтобы в функцию <code>baz</code> можно было передать функции вида <code>foo</code> и <code>bar</code>? 2. Пусть есть функции вида <code>double** foo();</code> и <code>void* bar(int*, char*);</code> Как синтаксически описать формальные аргументы функции <code>void baz(____, ____)</code> так, чтобы в функцию <code>baz</code> можно было передать функции вида <code>foo</code> и <code>bar</code>? 3. Пусть есть функции вида <code>int** foo(char*, double);</code> и <code>void bar();</code> Как для подобных функций ввести синонимы типов “указатель на функцию” с помощью ключевого слова <code>typedef</code>? Как потом синонимы использовать при описании формальных аргументов других функций? 4. Проведите сравнение реализованных вами методов на основании данных, полученных при тестировании, а также анализируя собственно реализации методов (в частности, возможные ограничения их применения).
--	--

Описание шкалы оценивания:

Оценка: зачтено

Описание характеристики выполнения знания: Оценка "зачтено" выставляется если задание выполнено правильно или с незначительными недочетами

Оценка: не зачтено

Описание характеристики выполнения знания: Оценка "не зачтено" выставляется если задание не выполнено в отведенный срок или результат не соответствует заданию

КМ-15. (Тест №2) Динамические структуры данных

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Тестирование

Вес контрольного мероприятия в БРС: 7

Процедура проведения контрольного мероприятия: В текстовом редакторе через интернет. Расчетное время 20 минут, для вычислений можно использовать технические средства. Варианты заданий вычисляются по номеру студента в ведомости БАРС. Результаты сохраняются в ОСЭП МЭИ (либо в тестовой системе Cisco Webex)

Краткое содержание задания:

Работа ориентирована на проверку знаний по теме “Динамические структуры данных. Обзор методов построения и анализа алгоритмов” (часть темы)

Контрольные вопросы/задания:

Знать: основные принципы хранения, доступа и обработки данных в динамических структурах, базах данных, при поточном и многопоточном программировании	1.Что такое информационно-логическая структура данных? (2 балла) 2.Что такое динамическая структура данных? (1 балл) 3.Список - это информационно-логическая или динамическая структура данных? (2 балла) 4.Какие виды списков вы знаете? (1 балл) 5.Какие виды деревьев вы знаете? (2 балла) 6.Какие базовые операции для работы со стеком вы знаете? (2 балла) 7.Какие базовые операции для работы с очередью и деком вы знаете? (3 балла) 8.Какие виды обхода деревьев вы знаете? (3 балла) 9.Опишите алгоритм построения упорядоченного дерева поиска. Где применяют такие деревья? (3 балла) 10.Чем отличается дерево от графа? (1 балл)
--	--

Описание шкалы оценивания:*Оценка: 5**Нижний порог выполнения задания в процентах: 80**Описание характеристики выполнения знания: Оценка "отлично" выставляется если в результате ответов было набрано не менее 16-ти баллов**Оценка: 4**Нижний порог выполнения задания в процентах: 60**Описание характеристики выполнения знания: Оценка "хорошо" выставляется если в результате ответов было набрано не менее 12-ти баллов**Оценка: 3**Нижний порог выполнения задания в процентах: 50**Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если в результате ответов было набрано не менее 10-ти баллов***КМ-16. (Тест №3) Базы данных. Инструменты многопоточного программирования****Формы реализации:** Компьютерное задание**Тип контрольного мероприятия:** Тестирование**Вес контрольного мероприятия в БРС:** 8**Процедура проведения контрольного мероприятия:** В текстовом редакторе через интернет. Расчетное время 20 минут, для вычислений можно использовать технические средства. Варианты заданий вычисляются по номеру студента в ведомости БАРС. Результаты сохраняются в ОСЭП МЭИ (либо в тестовой системе Cisco Webex)**Краткое содержание задания:**

Работа ориентирована на проверку знаний по теме “Двоичные файлы. Базы данных. Многопоточное программирование” (часть темы)

Контрольные вопросы/задания:

Знать: основные принципы хранения, доступа и обработки данных в динамических	1.Какие основные понятия приняты в реляционных базах данных? (2 балла) 2.Что означает понятие “нормальная форма” в теории
--	--

<p>структурах, базах данных, при поточном и многопоточном программировании</p>	<p>баз данных? (2 балла) 3. Таблица находится в первой нормальной форме, если... (продолжите). (2 балла) 4. Какие связи существуют между таблицами базы данных? (1 балл) 5. Какие запросы могут возникать в базах данных? (1 балл) 6. Процесс — это...(продолжите). (2 балла) 7. Поток — это...(продолжите). (2 балла) 8. Многопоточность — это...(продолжите). (2 балла) 9. Поясните назначение таких средств синхронизации, как “семафор” и “критическая секция”. (3 балла) 10. Поясните назначение таких средства синхронизации, как “взаимоисключение” (или “замок”) и “событие”. (3 балла)</p>
--	--

Описание шкалы оценивания:

Оценка: 5

Нижний порог выполнения задания в процентах: 80

Описание характеристики выполнения знания: Оценка "отлично" выставляется если в результате ответов было набрано не менее 16-ти баллов

Оценка: 4

Нижний порог выполнения задания в процентах: 60

Описание характеристики выполнения знания: Оценка "хорошо" выставляется если в результате ответов было набрано не менее 12-ти баллов

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Оценка "удовлетворительно" выставляется если в результате ответов было набрано не менее 10-ти баллов

КМ-17. Защита расчетного задания №2 "Динамические структуры данных. Разработка собственного класса в Python. Запись и считывание объектов (работа с бинарными файлами). Реализация итерируемого объекта и генератора в классе"

Формы реализации: Компьютерное задание

Тип контрольного мероприятия: Расчетно-графическая работа

Вес контрольного мероприятия в БРС: 30

Процедура проведения контрольного мероприятия: Заранее определяется общее задание на группу (группа разбивается на бригады по 2-3 человека) для самостоятельной и коллективной работы в качестве домашнего задания. Необходимо реализовать программный код решения, сопроводить решение спецификацией-отчетом (один на бригаду). Защита состоит в демонстрации работоспособности программы на тестах (или с помощью контрольной формулы) в компьютерных классах. Необходимо ответить на контрольные вопросы преподавателя по теме работы. На защите каждый член бригады отвечает за определенную часть выполненной работы полностью

Краткое содержание задания:

1. Целью работы является:

- a. изучение внутреннего устройства различных динамических структур данных, построенных на основе узлов и связей между узлами по predetermined базовым правилам;
 - b. освоение понятия класса в Python и получение навыка разработки собственных классов данных;
 - c. умение работать в составе коллектива, вести совместную разработку проекта (данная работа является бригадной 2-3 человека).
3. Работа выполняется на языке программирования Python. В качестве информационного поля узла во всех типах структур можно взять вариант — «целое число со знаком».
4. Для получения оценки выше «хорошо» необходимо дополнить класс методами `__next__` и `__iter__`, продемонстрировать их работу.
5. **Оформить отчет** (один на бригаду):
- a. формулировка задачи;
 - b. определение моделируемой структуры данных, ее схема;
 - c. описание, реализованного класса: структура данных с указанием полей данных; базовые операции и программный код их реализующий;
 - d. полный код программы с комментариями;
 - e. результаты тестирования программы

Контрольные вопросы/задания:

<p>Уметь: программировать собственные классы для динамических объектов на основании принятых базовых операций в заданных структурах данных с применением их в прикладных задачах</p>	<p>1.Реализовать класс «ДЕК» на базе ссылочной структуры «Линейный двусвязный список» со всеми основными операциями для этого типа структуры и продемонстрировать их работу. Добавить метод «копия» — создание копии дека по правилу:</p> <ul style="list-style-type: none"> a. если исходный дек1 установлен в состоянии «голова-хвост» (прямое направление), то его копия2 – полная идентичность исходному (голова1=голова2, хвост1=хвост2); b. если исходный дек1 установлен в состоянии «хвост-голова» (обратное направление), то его копия2 – перевернутый дек и назначено, что хвост1=голова2, голова1=хвост2 <p>2.Реализовать класс «динамическая прямоугольная матрица» на базе встроеного типа данных <i>List</i> в Python. Реализовать методы класса:</p> <ul style="list-style-type: none"> a. создать пустую матрицу; b. создание пустой матрицы размера $N \times M$ (выделение памяти); c. добавление строки после последней; d. добавление столбца после последнего; e. получить (считать) размер матрицы; f. изменить, прочитать элемент по его индексам; g. найти элемент, удовлетворяющий условию поиска: «элемент в диапазоне от A до B», где A, B – изменяемые параметры метода <p>3.Реализовать класс «бинарное дерево» на базе нелинейного трехсвязного (или двусвязного) списка. Реализовать методы класса:</p> <ul style="list-style-type: none"> a. создать пустое дерево; b. создать дерево на основе потока данных (<i>префиксная</i> форма обхода в глубину – КЛП);
--	---

	<p>с. печать бинарного дерева по правилу ЛКП (инфиксная форма);</p> <p>d. проверка пустоты бинарного дерева;</p> <p>е. удаление бинарного дерева.</p> <p>Один из тестов для демонстрации:</p> <hr/> <p style="text-align: center;"><i>ввод ГБА**В**Д*Е**</i></p> <p style="text-align: center;"><i>вывод *А*Б*В*Г*Д*Е* (где * - пустая вершина)</i></p> <p style="text-align: center;"><i>или АБВГДЕ (без вывода пустых вершин)</i></p> <hr/>
--	---

Описание шкалы оценивания:

Оценка: зачтено

Описание характеристики выполнения знания: Оценка "зачтено" выставляется каждому члену бригады индивидуально, если задание выполнено правильно или с незначительными недочетами, даны ответы на вопросы преподавателя

Оценка: не зачтено

Описание характеристики выполнения знания: Оценка "не зачтено" выставляется если задание не выполнено в отведенный срок или результат не соответствует заданию, либо на защите студент не смог ответить на вопросы преподавателя по своей индивидуальной части работы

СОДЕРЖАНИЕ ОЦЕНОЧНЫХ СРЕДСТВ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

2 семестр

Форма промежуточной аттестации: Зачет с оценкой

Пример билета

МЭИ	БИЛЕТ № ____	Утверждаю: Зав. кафедрой « » июня 20 г.
	Кафедра ПМИИ Дисциплина Программирование и основы алгоритмизации Институт ИЭЭ	
<p>1. Способы импортирования модулей в Python.</p> <p>2. Написать программу на языке C для решения задачи со встроенными массивами. Использовать не менее одной функции для вычисления в массивах.</p> <p>Задача. Если в целочисленном массиве все элементы упорядочены по неубыванию значений элементов, изменить массив по правилу: вначале расположить все четные элементы, далее все остальные. Сохранение исходного взаимного порядка в частях массива необязательно. Выделить альтернативу</p> <p>Лектор _____</p>		

Процедура проведения

Проводится в устной форме по билетам в виде решения задачи и последующего устного ответа по вопросу теории. Время на подготовку ответа – 40 минут

1. Перечень компетенций/индикаторов и контрольных вопросов проверки результатов освоения дисциплины

1. Компетенция/Индикатор: ИД-1_{УК-1} Выполняет поиск необходимой информации, ее критический анализ и обобщает результаты анализа для решения поставленной задачи

Вопросы, задания

1. Написать программу на языке C для решения задачи со встроенными массивами. Использовать не менее одной функции для вычисления в массивах.

Задача. Дан упорядоченный по возрастанию целочисленный массив RF длины fs . Осуществить поиск ключа Key методом бинарного поиска и вывести ответ: “ключ $<Key>$ найден” или “ключ $<Key>$ не найден”, в зависимости от результата

2. Написать программу на языке C для решения задачи со встроенными массивами. Использовать не менее одной функции для вычисления в массивах.

Задача. Дан массив KB длины L из целых чисел. Осуществить сортировку массива по убыванию абсолютных значений методом вставки

3. Понятие вычислительной сложности алгоритма и оценки сложности

4. Дайте оценку сложности для алгоритма обменной сортировки. Почему разделяются такие действия, как “сравнение” и “обмен” (элементов) при вычислении сложности алгоритмов

Материалы для проверки остаточных знаний

1. Алгоритм - это...

Ответы:

- 1) абстрактная или реальная система, способная выполнить действия, предписанные исполнителю; 2) понятное и точное предписание исполнителю выполнить определенную последовательность действий для решения некоторой задачи за конечное число шагов; 3) инструкция для решения поставленной задачи

Верный ответ: 2

2. Свойство алгоритма "дискретность" означает, что...

Ответы:

1) алгоритм должен представлять процесс решения задачи как последовательность отдельных действий; 2) каждое правило алгоритма должно быть однозначным и не оставлять места для произвольного толкования; 3) исполнитель точно знает, какое действие выполнить следующим

Верный ответ: 1

3. Укажите базовые алгоритмические структуры (возможно несколько ответов):

Ответы:

1) действие; 2) массив; 3) переменная; 4) ветвление; 5) цикл

Верный ответ: 1, 4, 5

4. Укажите виды циклов

Ответы:

1) для нахождения суммы, для обработки массивов; 2) с условием, с заданным числом повторений; 3) для ввода данных в программу, для обработки значений элементов массива

Верный ответ: 2

5. Цикл - это

Ответы:

1) алгоритмическая структура, которая содержит проверку некоторого условия; 2) алгоритмическая структура, которая обеспечивает многократное выполнение некоторой совокупности действий; 3) оследовательность действий, следующих одно за другим

Верный ответ: 2

6. К целочисленному типу данных относятся следующие типы данных:

Ответы:

1) integer, longint, string; 2) integer, longint, real; 3) integer, byte, word

Верный ответ: 3

7. Если файл открыт с аббревиатурой "rt", то это означает режим открытия...

Ответы:

1) текстового файла в режиме считывания; 2) текстового файла в режиме записи; 3) двоичного файла в режиме считывания; 4) текстового файла в режиме считывания и записи

Верный ответ: 1

8. Какие понятия относятся к свойствам сортировок:

Ответы:

1) устойчивость, естественность; 2) правильность, независимость

Верный ответ: 1

2. Компетенция/Индикатор: ИД-1_{ОПК-1} Применяет средства информационных, компьютерных и сетевых технологий для поиска, хранения, обработки, анализа и представления информации

Вопросы, задания

1. Написать программу с использованием нескольких функций на языке Python для решения следующей задачи:
исходный текстовый файл содержит слова, разделенные символом “запятая”. Необходимо выполнить сортировку слов в алфавитном порядке и записать в новый текстовый файл с разделителем “пробел”. Для сортировки использовать метод обменной сортировки
2. Написать программу с использованием нескольких функций на языке Python для решения следующей задачи:
исходный текстовый файл содержит слова, набранные в произвольном регистре (одно слово может совмещать оба регистра, например, “Яблоны”), разделенные символом “пробел”. Необходимо выполнить перекодирование исходного файла в файл, в котором все слова будут набраны в нижнем регистре, причем каждое слово отделено от другого разделителем “;”
3. Укажите два основных способа подключения внешних библиотек в Python
4. Какие внешние и внутренние библиотеки для языка Python вам известны?
5. Опишите основные методы модуля *random* для Python
6. Опишите основные методы модуля *time* для Python
7. Написать программу на языке Python с использованием методов модуля *random* для решения следующей задачи:
необходимо сгенерировать случайный пароль заданной длины *D*, содержащий символы “цифр”, “букв” латинского алфавита и специальные символы “(”, “_”, “)”. Пароль записать в текстовый файл

Материалы для проверки остаточных знаний

1. Какие числа будут выведены на экран?

```
for t in range(-7, -3, -1):
```

```
    print(t)
```

Ответы:

- 1) -3, -4, -5, -6, -7; 2) никакие, так как **range()** не работает с отрицательными числами; 3) никакие, так как нарушен диапазон для отрицательного шага: $-7 < -3$

Верный ответ: 3

2. Какие числа будут выведены на экран?

```
for t in range(-7, -3, 1):
```

```
    print(t)
```

Ответы:

1) -7, -6, -5, -4, -3; 2) -7, -6, -5, -4; 3) никакие, так как **range()** не работает с отрицательными числами;

Верный ответ: 2

3. Выберите правильный вариант выделения динамической памяти под неименованную переменную типа *float*

Ответы:

1. 1) `float *ptr = new float;` 2) `float & ptr = new float;` 3) `float * ptr = &X; X = new float;`

Верный ответ: 1

3. Компетенция/Индикатор: ИД-1опк-2 Алгоритмизирует решение задач и реализует алгоритмы с использованием программных средств

Вопросы, задания

1. Структура программы в С (С++)

2. Понятие «приоритет операции», свойство ассоциативности операции. Пример, демонстрирующий влияние приоритета и ассоциативности

3. Область видимости в С (С++)

4. Пространство имен, именованные пространства в С++ (способы доступа к членам пространства имен на примере реализации стека)

5. Область видимости и пространство имен в Python

6. Написать программу на языке С для решения задачи со встроенными массивами.

Использовать не менее одной функции для вычисления в массивах. Обеспечить ввод массива данными с клавиатуры (поток ввода) и вывести на экран (поток вывода).

Задача. Дан массив *A* длины *g* из положительных вещественных чисел. Определить наименьшее из значений $\sin(A[i]) + (A[i] - A[i-1])^2$. Вывод результата обеспечить в форматном виде с заданным *epsilon* количеством знаков после запятой

Материалы для проверки остаточных знаний

1.

```
while условие do
    оператор;
```

Как называется такой цикл?

Ответы:

- 1) цикл со счетчиком; 2) цикл с заданным числом повторений; 3) цикл с предусловием

Верный ответ: 3

2. Назовите используемый ниже оператор

```
if (a < b) {
    min = a;
} else {
    min = b;
```

```
}
```

Ответы:

1) условный оператор, полная форма; 2) условный оператор, неполная форма; 3) оператор выбора

Верный ответ: 1

3. Что делает данный фрагмент программы?

```
sum = 0;
```

```
for (t = 1; t < 8; t++){
```

```
sum = sum * t;
```

```
}
```

Ответы:

1) находит сумму первых семи натуральных чисел; 2) находит произведение первых семи натуральных чисел; 3) в программе ошибка

Верный ответ: 3

II. Описание шкалы оценивания

Оценка: 5

Нижний порог выполнения задания в процентах: 80

Описание характеристики выполнения знания: Работа выполнена в рамках "продвинутого" уровня. Ответы даны верно, четко сформулированные особенности практических решений

Оценка: 4

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Работа выполнена в рамках "базового" уровня. Большинство ответов даны верно. В части материала есть незначительные недостатки

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Работа выполнена в рамках "порогового" уровня. Основная часть задания выполнена верно. На вопросы углубленного уровня даны неверные ответы, либо ответы со значительными ошибками

III. Правила выставления итоговой оценки по курсу

Оценка определяется в соответствии с Положением о балльно-рейтинговой системе для студентов НИУ «МЭИ» на основании семестровой и аттестационной составляющих

3 семестр

Форма промежуточной аттестации: Экзамен

Пример билета

МЭИ	ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № ____	Утверждаю: Зав. кафедрой
	Кафедра ПМИИ Дисциплина Программирование и основы алгоритмизации Институт ИЭЭ	«__» января __20__ г.
<p>1. Статическая и динамическая типизация. Основные данные, связываемые с объектом в Python (счетчик ссылок, тип, значение, идентификатор). Функции <code>type()</code> и <code>isinstance()</code>, назначение, пример использования.</p> <p>2. Динамические структуры. Связные линейные списки (одно и двунаправленные).</p> <p>3. Написать программу на языке C для решения задачи с динамическими массивами. Использовать функции для выделения (освобождения) памяти под массив, ввода массива, вывода массива на консоль, не менее одной функции для вычисления в массивах. Задача:</p> <p>если в целочисленной квадратной матрице на главной диагонали все элементы четны, сформировать новый массив, состоящий из элементов, расположенных над главной диагональю матрицы (обход матрицы слева направо, сверху вниз). Выделить альтернативу.</p> <p>Лектор _____</p>		

Процедура проведения

Проводится в устной форме по билетам в виде подготовки и изложения развернутого ответа. Время на подготовку ответа – 90 минут

1. Перечень компетенций/индикаторов и контрольных вопросов проверки результатов освоения дисциплины

1. Компетенция/Индикатор: ИД-1_{УК-1} Выполняет поиск необходимой информации, ее критический анализ и обобщает результаты анализа для решения поставленной задачи

Вопросы, задания

1. Написать программу на языке Python для решения задачи, где одна функция используется в качестве параметра другой функции

Написать подпрограммы ($F1$ и $F2$), изменяющие состояние исходного одномерного целочисленного массива по указанному правилу: $F1$ — все отрицательные элементы должны сместиться в конец массива (порядок сохранен); $F2$ — все отрицательные элементы должны быть заменены на квадраты их значений.

Написать подпрограмму ($F3$), принимающую на вход два целочисленных массива, целочисленный параметр P , подпрограмму изменения массива и выполняет следующие действия: изменяет массивы; объединяет массивы (новый массив); находит номер максимального элемента в новом массиве и количество элементов в нем, равных P . Возвращает новый массив, найденный номер и количество.

Применить $F3$ последовательно к следующим наборам параметров: $[A(n), B(m), F1, -35]$ и $[A(n), C(s), F2, 33]$.

Figure 1 Задача

2. Способы передачи параметров в подпрограмму: по значению, по ссылке
3. Передача функции в качестве параметра другой функции в Python. Привести пример
4. Возвращаемые значения функций в C (C++) (правила, небезопасные операции)
5. Написать программу на языке Python для решения задачи, где одна функция используется в качестве параметра другой функции

Написать подпрограммы ($F1$ и $F2$), вычисляющие функции на выборочных элементах заданного одномерного вещественного массива по указанному правилу: $F1$ — поиск максимума на элементах массива, начиная с номера $k1$ по номер $k2$; $F2$ — поиск минимума на элементах массива, начиная с номера $d1$ по номер $d2$.
 Написать подпрограмму ($F3$), принимающую на вход массив $A(m)$, подпрограмму ($F1$) вычисления над элементами массива и выполняет следующие действия: вычисляет значение по формуле $Y = m \cdot M_1 + (m - 1) \cdot M_2 + \dots + 2 \cdot M_{m-1} + 1 \cdot M_m$, где M_j — значение, полученное из подпрограммы ($F1$) для выборочной части массива с номерами от 1 до j , соответственно. Возвращает найденное значение.
 Применить $F3$ последовательно к следующим наборам заданных параметров: $[A(m), F1]$, $[B(s), F2]$, $[A(m), F2]$ и $[C(d), F1]$.

Figure 2 Задача

Материалы для проверки остаточных знаний

1. Какие парадигмы и стили программирования поддерживает Python? (возможно несколько ответов)

Ответы:

1) логическое программирование; 2) объектно-ориентированное; 3) модульное программирование; 4) императивный подход

Верный ответ: 2, 3, 4

2. К каким типам относятся файлы CSV?

Ответы:

1) текстовый; 2) двоичный; 3) типизированный; 4) бестиповой

Верный ответ: 1

2. Компетенция/Индикатор: ИД-1_{ОПК-1} Применяет средства информационных, компьютерных и сетевых технологий для поиска, хранения, обработки, анализа и представления информации

Вопросы, задания

1. Синтаксис определения класса в Python. Объект-класс и объект-экземпляр

2. Написать программу на языке C для решения задачи с динамическими массивами.

Использовать функции для выделения (освобождения) памяти под массив, ввода массива, вывода массива на консоль, не менее одной функции для вычисления в массивах

Дана прямоугольная целочисленная матрица $D(n, m)$. Сформировать массив B , состоящий из всех элементов матрицы D кратных числу k . Обход матрицы выполнять по столбцам, начиная с последнего (сверху вниз и справа налево). Выделить альтернативу.

Figure 3 Задача

3. Написать программу на языке C для решения задачи с динамическими массивами.

Использовать функции для выделения (освобождения) памяти под массив, ввода массива, вывода массива на консоль, не менее одной функции для вычисления в массивах

Для заданных прямоугольных целочисленных матриц $D(n, m)$ и $B(m, n)$ выполнить матричные операции: $(D \cdot B)^T$.

Figure 4 Задача

4. Модуль расширения работы с БД. Объект-соединение, объект-курсор, объекты-типы

5. Основные понятия реляционной СУБД. Что такое DB API 2.0

6. Работа с потоками в Python с помощью модуля **threading** (способ: «функциональный», «классовый»)

7. Многопоточность в Python, понятия: процесс, поток, синхронизация

8. Динамические структуры. Связные списки (деревья). Виды обхода деревьев

9. Динамические структуры. Связные линейные списки (одно и двунаправленные)

10. Динамические структуры. Связные линейные списки (стек, очередь, дек)

11. Концепция ООП, основные понятия. Понятие класса и объекта класса в общей методологии

12. Пример создания собственного итератора на базе класса с реализацией **next()** и **iter()**

Материалы для проверки остаточных знаний

1. Укажите верное утверждение для “информационно-логическая структура СТЕК” может быть реализована с помощью:

Ответы:

1) динамического массива; 2) статического массива (неизменяемый размер); 3) динамического списка; 4) двоичного файла; 5) любой тип, указанный в предыдущих пунктах

Верный ответ: 5

2. Как называется свойство системы, позволяющее описать новый класс на основе уже существующего с частично или полностью заимствующейся функциональностью?

Ответы:

1) абстракция; 2) инкапсуляция; 3) наследование; 4) полиморфизм

Верный ответ: 3

3. Как называется свойство системы, позволяющее использовать объекты с одинаковым интерфейсом без информации о типе и внутренней структуре объекта?

Ответы:

1) абстракция; 2) инкапсуляция; 3) наследование; 4) полиморфизм

Верный ответ: 4

4. Как называется свойство системы, позволяющее объединить данные и методы, работающие с ними, в единой синтаксической конструкции?

Ответы:

1) абстракция; 2) инкапсуляция; 3) наследование; 4) полиморфизм

Верный ответ: 2

5. Для эффективной работы БД должно выполняться условие:

Ответы:

1) достоверности данных; 2) объективности данных; 3) непротиворечивости данных

Верный ответ: 3

6. Основное отличие реляционной БД?

Ответы:

1) данные организовываются в виде отношений (связей); 2) строго древовидная структура; 3) представлена в виде графов

Верный ответ: 1

3. Компетенция/Индикатор: ИД-1_{ОПК-2} Алгоритмизирует решение задач и реализует алгоритмы с использованием программных средств

Вопросы, задания

1. Написать программу на языке Python для решения задачи с применением подходящих функций для обработки последовательностей (map, zip, reduce, filter, lambda, генераторы коллекций)

Задан текстовый файл с литературным содержанием. Необходимо считать текст по заданному правилу и произвести его статистический анализ.

Правило: разбить текст на отдельные предложения (окончанием предложения считать символы «.», «!» и «?»), и разместить в списке (предложение – это элемент списка). Обработать список, вычисляя для каждого его элемента и формируя соответствующий кортеж со следующей информацией: а) число слов в предложении; б) количество знаков, исключая пробелы и запятые; в) количество всех знаков в предложении. Записать все полученные кортежи в текстовый и бинарный файлы.

Figure 5 Задача

2. Написать программу на языке C для решения рекурсивной задачи в динамическом массиве. Использовать функции для выделения (освобождения) памяти под массив, ввода массива, вывода массива на консоль, рекурсивную функцию вычислений

Для заданного одномерного массива B из N элементов с помощью рекурсивной функции найти номер минимального значения выражения $B_i \sin B_i^2$. В рекурсивной функции каждый раз отделять последнюю треть от первых двух третей рассматриваемой части массива и применять эту же функцию к обеим частям. Рекурсивные вызовы заканчивать, когда останется только один элемент в рассматриваемой части.

Например, для $N = 9$: $\min_{i=1}^9 (B_i \sin B_i^2) = \min (\max_{i=1}^3 (B_i \sin B_i^2); \min_{i=7}^9 (B_i \sin B_i^2))$.

Figure 6 Задача

3. Написать программу на языке C для решения рекурсивной задачи в динамическом массиве. Использовать функции для выделения (освобождения) памяти под массив, ввода массива, вывода массива на консоль, рекурсивную функцию вычислений

Для заданного одномерного массива A из N элементов с помощью рекурсивной функции найти количество элементов массива, для которых выполняется условие $\sin A_i < \cos A_i$. В рекурсивной функции каждый раз отделять последнюю треть от первых двух третей рассматриваемой части массива и применять эту же функцию к обеим частям. Рекурсивные вызовы заканчивать, когда останется только один элемент в рассматриваемой части массива. Например, для $N = 12$:
 $Count_{i=1}^{12}(\sin A_i < \cos A_i) = Count_{i=1}^8(\sin A_i < \cos A_i) + Count_{i=9}^{12}(\sin A_i < \cos A_i)$.

Figure 7 Задача

4. Работа с бинарными файлами. Модуль **Pickle**

5. Оператор **with**. Файлы в формате CSV (устройство и пример работы)

6. Написать программу на языке Python для решения задачи с применением подходящих функций для обработки последовательностей (**map**, **zip**, **reduce**, **filter**, **lambda**, генераторы коллекций)

Задан текстовый файл, состоящий из потока вещественных чисел, разделенных знаком «,». Необходимо считать «числа» в список с преобразованием к вещественному типу и отсортировать любым известным методом сортировки. Сформировать кортеж со следующей информацией о списке: (максимальное значение, минимальное значение, число максимальных элементов, число минимальных элементов, число положительных элементов, число элементов, равных нулю). Записать в бинарный файл кортеж с информацией о списке и отсортированный список.

Figure 8 Задача

7. Взаимодействие программ с вызывающим контекстом в Python (вызов по соиспользованию). Примеры

8. Рекурсия и определения: простая и косвенная рекурсия, рабочая и терминальная ветвь рекурсивного алгоритма, база рекурсии и шаг рекурсии. Пояснить примером

9. Рекурсия и определения: прямой и обратный ход рекурсии, префиксная и постфиксная запись рекурсии, индикатор завершения рекурсивных вызовов и декомпозиция.

Пояснить примером

10. Сравнение переменных в C и Python

11. Понятие хеш (хеш-функция) и уникальный идентификатор. Функции **hash()** и **id()**, назначение, пример использования

12. Подсчет ссылок и сборка мусора в Python, функция **getrefcount()**

13. Элементы функционального программирования в Python: генераторы списков

14. Элементы функционального программирования в Python: функции **map()**, **reduce()**, **filter()** и **zip()**

15. Типы генераторов. Генератор-функция

Материалы для проверки остаточных знаний

1. Какие способы для объявления ссылок правильны? (возможно несколько ответов)

Ответы:

1) `int s; int& rs=s;` 2) `int& rs;` 3) `int a=3; int &rs=&a;` 4) `int a; int &rs=a; int &rss=rs;`

Верный ответ: 1, 4

2. Какой результат будет получен после исполнения следующих инструкций?

`List1 = [1, 2, 3]`

`List2 = [4, 5, 6]`

`List3 = [-10]`

```
print ( list( map( lambda x, y, z: x + y + z, List1, List2, List3) ) )
```

Ответы:

1) приведет к ошибке; 2) [-5]; 3) [-5, 7, 9]

Верный ответ: 2

3.Какая последовательность будет выведена на экран в результате исполнения следующих инструкций?

```
def rec (n):  
    if n>0:  
        rec(n-1)  
    print(n, end=' ')
```

```
rec(5)
```

Ответы:

1) 1 2 3 4 5; 2) 5, 4, 3, 2, 1; 3) ничего не будет выведено

Верный ответ: 1

4.Что произойдёт при попытке исполнения инструкций?

```
def rec (n):  
    if n > 0:  
        return rec (n % 10)  
    else:  
        return n
```

```
a = rec (15)
```

Ответы:

1) значение a станет равно 5; 2) переполнение программного стека; 3) заикливание; 4) значение станет равно 0

Верный ответ: 2

5.Выберете правильную (возможно несколько ответов) синтаксически запись прототипа функции **bar**, если функцию вида **int** foo()**; необходимо передать в функцию **int bar(...)**; в качестве ее единственного аргумента:

Ответы:

1) **int bar(int (**foo))**; 2) **int bar(int** (*foo)())**; 3) **int bar(int** (foo)())**; 4) **int bar(int** (*baz)())**;

Верный ответ: 2, 4

II. Описание шкалы оценивания

Оценка: 5

Нижний порог выполнения задания в процентах: 90

Описание характеристики выполнения знания: Работа выполнена в рамках "продвинутого" уровня. Выставляется студенту, правильно выполнившему практическое задание, который показал при ответе на вопросы экзаменационного билета и на дополнительные вопросы, что владеет материалом изученной дисциплины, свободно применяет свои знания для объяснения различных явлений и решения задач

Оценка: 4

Нижний порог выполнения задания в процентах: 70

Описание характеристики выполнения знания: Работа выполнена в рамках "базового" уровня. Выставляется студенту, правильно выполнившему практическое задание и в основном правильно ответившему на вопросы экзаменационного билета и на дополнительные вопросы, но допустившему при этом не принципиальные ошибки

Оценка: 3

Нижний порог выполнения задания в процентах: 50

Описание характеристики выполнения знания: Работа выполнена в рамках "порогового" уровня. Выставляется студенту, который в ответах на вопросы экзаменационного билета допустил существенные и даже грубые ошибки, но затем исправил их сам, а также не выполнил практическое задание из экзаменационного билета, но либо наметил правильный путь его выполнения, либо по указанию экзаменатора решил другую задачу из того же раздела дисциплины

III. Правила выставления итоговой оценки по курсу

Оценка определяется в соответствии с Положением о балльно-рейтинговой системе для студентов НИУ «МЭИ» на основании семестровой и аттестационной составляющих