

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Гаскаев Сергей Валерьевич
Должность: Ректор
Дата подписания: 15.06.2026 12:17:24
Уникальный программный ключ:
04c19ed8bfb985b6cb7724869a8788b327573



МИНОБРАЗОВАНИЯ И НАУКИ РОССИИ
Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Челябинский государственный университет» (ФГБОУ ВО «ЧелГУ»)

Фонд оценочных средств для промежуточной аттестации по дисциплине «Технология программирования на языке C++» по направлению подготовки 02.03.01 «Математика и компьютерные науки» направленности «Математические и компьютерные методы в фундаментальных и прикладных исследованиях»
ФГБОУ ВО «ЧелГУ»

стр. 1

**Фонд оценочных средств для промежуточной аттестации по дисциплине (модулю)
«Технология программирования на языке C++»**

**Направление подготовки (специальность)
02.03.01 «Математика и компьютерные науки»**

**Направленность (профиль)
«Математические и компьютерные методы в фундаментальных и прикладных исследованиях»**

**Присваиваемая квалификация
Бакалавр**

**Форма обучения
Очная**

**Год набора
2026**

Челябинск, 2026 г.



Содержание

1. Паспорт фонда оценочных средств	3
2. Перечень формируемых компетенций	4
3. Содержание оценочных средств по дисциплине	6
3.1. Виды оценочных средств	6
3.2. Содержание оценочных средств	7
4. Порядок проведения и критерии оценивания промежуточной аттестации	19
4.1. Порядок проведения промежуточной аттестации	19
4.2. Критерии оценивания промежуточной аттестации по видам оценочных средств	19
4.3. Результаты промежуточной аттестации и уровни сформированности компетенций.....	20



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Челябинский государственный университет» (ФГБОУ ВО «ЧелГУ»)

Фонд оценочных средств для промежуточной аттестации по дисциплине «Технология программирования на языке C++» по направлению подготовки 02.03.01 «Математика и компьютерные науки» направленности «Математические и компьютерные методы в фундаментальных и прикладных исследованиях»
ФГБОУ ВО «ЧелГУ»

стр. 3

1. Паспорт фонда оценочных средств

Направление подготовки: 02.03.01 Математика и компьютерные науки.

Направленность: Математические и компьютерные методы в фундаментальных и прикладных исследованиях.

Дисциплина: Технология программирования на языке C++.

Семестры: 1, 2.

Форма промежуточной аттестации: зачёт в 1 семестре, экзамен во 2 семестре.

Для оценивания результатов обучения используется балльно-рейтинговая система.



2. Перечень формируемых компетенций

Изучение дисциплины «Технология программирования на языке C++» направлено на формирование компетенций, приведённых в Таблице 1.

Таблица 1. Результаты обучения по дисциплине.

Код и наименование компетенции согласно ФГОС	Индикаторы достижения компетенций согласно ОПОП ВО	Перечень планируемых результатов обучения по дисциплине
1	2	3
ОПК-4: Способен находить, анализировать, реализовывать программно и использовать на практике математические алгоритмы, в том числе с применением современных вычислительных систем	ОПК-4.1. Демонстрирует знание теории алгоритмов, методологии и технологии программирования. ОПК-4.2. Демонстрирует умения находить, анализировать, реализовывать программно математические алгоритмы, в том числе с применением современных вычислительных систем. ОПК-4.3. Имеет практический опыт программной реализации математических алгоритмов.	Знать основные методы сбора и анализа информации, способы формализации цели и методы ее достижения. Уметь анализировать, обобщать и воспринимать информацию; ставить цель и формулировать задачи по её достижению. Владеть основами программирования, навыками работы в сети.
ОПК-6: Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ОПК-6.1. Демонстрирует знание основ технологий программирования и базисных алгоритмов. ОПК-6.2. Демонстрирует умения разрабатывать алгоритмы и компьютерные программы. ОПК-6.3. Имеет практические навыки разработки компьютерных программ.	Знать основы теории алгоритмов и ее применения; синтаксис, семантику и формальные способы описания алгоритмов; основные структуры данных, механизмы их реализации и методы работы с ними; основные методы поиска и сортировки данных. Уметь оценивать объемы обрабатываемой информации и вычислительную сложность алгоритмов; выбирать соответствующие структуры для организации данных. Владеть базовыми методами работы со структурами данных, эффективными способами поиска и сортировки данных.
ПК-2: Способен использовать базовые математические знания и информационные технологии при проектировании программного обеспечения	ПК-2.1. Обладает знаниями о существующих типовых шаблонах проектирования программного обеспечения; о методах и средства проектирования программного обеспечения, структурах данных, баз данных. ПК-2.2. Демонстрирует умения: применять типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения; применять методы и средства проектирования программного обеспечения, баз данных, структур данных; применять методы и средства создания программного обеспечения. ПК-2.3. Имеет практический опыт (навыки): применения стандартных алгоритмов при проектировании и	Знать современные IDE. Уметь использовать современные IDE. Владеть программированием в современных IDE.



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Челябинский государственный университет» (ФГБОУ ВО «ЧелГУ»)

Фонд оценочных средств для промежуточной аттестации по дисциплине «Технология программирования на языке C++» по направлению подготовки 02.03.01 «Математика и компьютерные науки» направленности «Математические и компьютерные методы в фундаментальных и прикладных исследованиях» ФГБОУ ВО «ЧелГУ»

стр. 5

создании программного обеспечения;
разработки и реализация алгоритмов
решения задач в соответствии с
поставленными условиями; использования
методов и приемов алгоритмизации
поставленных задач.



3. Содержание оценочных средств по дисциплине

3.1. Виды оценочных средств

Таблица 2. Виды оценочных средств.

Код, наименование компетенции согласно ФГОС	Перечень планируемых результатов обучения по дисциплине	Контролируемые темы/разделы (номер и название раздела из РПД п.2.2)	Семестр	Номер задания	Наименование оценочного средства
ОПК-4: Способен находить, анализировать, реализовывать программно и использовать на практике математические алгоритмы, в том числе с применением современных вычислительных систем	Знать основные методы сбора и анализа информации, способы формализации цели и методы ее достижения.	Основные конструкции программирования Алгоритмы и структуры данных	1-2	1-4	Контрольная работа
	Уметь анализировать, обобщать и воспринимать информацию; ставить цель и формулировать задачи по её достижению.			1-14	Тест
	Владеть основами программирования, навыками работы в сети.				
ОПК-6: Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	Знать основы теории алгоритмов и ее применения; синтаксис, семантику и формальные способы описания алгоритмов; основные структуры данных, механизмы их реализации и методы работы с ними; основные методы поиска и сортировки данных.			1-33	Вопросы к зачету
	Уметь оценивать объемы обрабатываемой информации и вычислительную сложность алгоритмов; выбирать соответствующие структуры для организации данных.			1-25	Вопросы к экзамену
	Владеть базовыми методами работы со структурами данных, эффективными способами поиска и сортировки данных.				
ПК-2: Способен использовать базовые	Знать современные IDE.			1-4	Задачи к экзамену
	Уметь использовать современные IDE.				



математические знания и информационные технологии при проектировании программного обеспечения	Владеть программирования в современных IDE.				
---	---	--	--	--	--

Типовые задания, критерии и показатели оценивания в рамках текущего контроля представлены в рабочей программе дисциплины (модуля). Полные комплекты оценочных средств и контрольно-измерительных материалов хранятся на кафедре.

3.2. Содержание оценочных средств

Промежуточная аттестация проводится в виде зачёта в 1 семестре, зачёта и экзамена во 2 семестре.

Оценочные средства представлены набором задач для проведения контрольных работ, выполняемых студентами в заданные сроки, тестом и набором экзаменационных билетов, предполагающих письменный ответ, оцениваемый преподавателем в присутствии экзаменуемого.

3.2.1. Задачи для проведения контрольных работ

№ п/п	Текст задачи	Номер на сайте
Контрольная работа №1		
1	<p>Ферзь может перемещаться на любое число полей по вертикали, горизонтали и диагонали (при условии, что на его пути нет фигур). Дана позиция ферзя на пустой бесконечной доске, необходимо определить минимальное количество ходов, необходимое ферзю для достижения новой позиции на доске.</p> <p><i>Формат ввода</i></p> <p>Ввод содержит четыре целых числа X_1, Y_1, X_2 и Y_2 ($-10^9 < X_1, Y_1, X_2, Y_2 < 10^9$). Ферзь находится на клетке (X_1, Y_1) и должен добраться до клетки (X_2, Y_2). Колонки на доске нумеруются слева направо, а строки – снизу вверх. Клетка в колонке X и строке Y имеет координаты (X, Y).</p> <p><i>Формат вывода</i></p> <p>Ваша программа должна вывести одно целое число – минимальное количество ходов ферзем от начальной позиции на доске до конечной.</p>	irc-1720
2	<p>Компания из M человек пришла в пиццерию. Посоветовавшись, они решили заказать одну большую пиццу с K начинками. Пицца представляет собой круг, поделённый на K равных секторов, в каждом из которых находится своя начинка. Пиццу подают ещё не разрезанной.</p> <p>Друзья попросили официанта разрезать пиццу на M равных секторов, по одному куску на человека, так, чтобы как можно большему количеству людей достался кусок по крайней мере с двумя начинками.</p> <p>Помогите официанту определить, какому именно количеству людей достанется больше одной начинки, если резать пиццу наиболее оптимально.</p> <p>Вводятся два целых числа K, M ($1 \leq K \leq 100, 1 \leq M \leq 100$) – количество начинок в пицце и количество человек в компании соответственно.</p>	irc-1763



	Выведите количество человек, которым достанется более одной начинки в наилучшем случае.	
3	<p>Выходя из школьного автобуса, Лиза не заметила, что Нельсон подставил ей подножку. Собрав разлетевшиеся при падении страницы своего доклада, Лиза их пересчитала и обнаружила, что одна из страниц пропала.</p> <p>Напишите программу, которая поможет Лизе определить номер потерянной страницы.</p> <p>Первая строка ввода содержит одно целое число — количество страниц в докладе N ($1 \leq N \leq 30000$). Далее следует $N-1$ строка, каждая из которых содержит одно целое число от 1 до N – номера собранных страниц. Все номера страниц различны.</p> <p>Вывести одно целое число – номер потерянной страницы.</p>	ipc-1499
4	<p>В магазине Апу стоит большая стеклянная банка с конфетами N сортов. Гомер схватил горсть конфет из банки, но узкое горлышко банки помешало ему вытащить руку. Гомер хочет вытащить из банки как минимум K конфет одного сорта, неважно какого. Напишите программу, определяющую, какое минимальное количество конфет Гомер должен оставить в своей руке, чтобы добиться поставленной цели.</p> <p>Первая строка ввода содержит два целых числа - количество сортов N и количество одинаковых конфет K ($1 \leq N, K \leq 10$).</p> <p>Вывести одно число — минимальное количество конфет, среди которых будет не менее K конфет одного сорта.</p>	ipc-1505
5	<p>Из города Китеж в форт Байан почти каждый час с 7 до 17 отправляются автобусы с туристами. К сожалению, в расписании указано только время отправления, но не указано время прибытия. Петя узнал, что время поездки до форта без остановок равно K часов. Но согласно постановлению главного врача Беловодья водители во время поездки должны делать остановки с 8 до 9 на завтрак, с 13 до 14 на обед и с 18 до 19 на ужин.</p> <p>Напишите программу, которая поможет Пете узнать время прибытия автобуса.</p> <p>Первая строка ввода содержит два целых числа – время отправления автобуса T ($7 \leq T \leq 17, T \neq 8, T \neq 13$) и длительность поездки без остановок K ($2 \leq K \leq 10$).</p> <p>Вывести одно целое число – время прибытия автобуса с учетом остановок на еду. Входные данные таковы, что автобусы прибывают в форт не позднее 23.</p>	ipc-1519
Контрольная работа №2		
1	<p>Рассмотрим последовательность целых чисел A_1, A_2, \dots, A_n. Парой будем называть два числа этой последовательности A_i и A_j, в которой $1 \leq i < j \leq n$ и $A_i = A_j$. Напишите программу, определяющую количество пар в последовательности.</p> <p>Ввод</p> <p>В первой строке содержится одно целое число n ($1 \leq n \leq 100000$) – количество чисел в последовательности. Далее следует n строк, содержащих по одному целому числу в диапазоне от -10^9 до 10^9 – элементы последовательности.</p> <p>Вывод</p> <p>В первой строке вывести количество пар в данной последовательности.</p>	ipc-107
2	<p>В первой строке ввода содержится целое число N ($1 \leq N \leq 100000$) – количество чисел в последовательности. Далее следует N строк, в каждой строке одно целое число из последовательности в диапазоне от 0 до 10^9.</p>	ipc-174



	Вывести количество различных чисел во введенной последовательности.	
3	<p>Джон составляет график дежурств в общежитии колледжа. Джон учится на дизайнера и поэтому решил упорядочить имена в списке по их длине, самые короткие имена – в начале списка, самые длинные – в конце, чтобы на доске объявлений список выглядел красиво.</p> <p>Напишите программу, которая поможет Джону расположить имена в нужном порядке.</p> <p>В первой строке ввода содержится положительное число N ($1 \leq N \leq 100$), далее следует N строк, в каждой строке одно имя. Все имена имеют различную длину, не превышающую 100 символов, и состоят только из латинских букв.</p> <p>Программа должна выдать упорядоченный по длине список имен.</p>	ipс-731
4	<p>Саша и Катя учатся в начальной школе. Для изучения арифметики при этом используются карточки, на которых написаны цифры (на каждой карточке написана ровно одна цифра). Однажды они пришли на урок математики, и Саша, используя все свои карточки, показал число A, а Катя показала число B. Учитель тогда захотел дать им такую задачу, чтобы ответ на нее смогли показать и Саша, и Катя, каждый используя только свои карточки. При этом учитель хочет, чтобы искомое число было максимально возможным.</p> <p>Во входном файле записано два целых неотрицательных числа A и B (каждое число в одной строке). Длина каждого из чисел не превосходит 100 000 цифр.</p> <p>Выведите одно число – максимальное целое число, которое можно составить, используя как цифры первого числа, так и цифры второго числа. Если же ни одного такого числа составить нельзя, выведите -1.</p>	ipс-1726
5	<p>Вася готовится к олимпиаде. Учитель дал ему N задач для тренировки. Для каждой из этих задач известно, каким умением a_i нужно обладать для её решения. Это означает, что если текущее умение Васи больше либо равно заданного умения для задачи, то он может ее решить. Кроме того, после решения i-й задачи Васино умение увеличивается на число b_i.</p> <p>Исходное умение Васи равно A. Решать данные учителем задачи он может в произвольном порядке. Какое максимальное количество задач он сможет решить, если выберет самый лучший порядок их решения?</p> <p>Сначала вводятся два целых числа N, A ($1 \leq N \leq 100\ 000$, $0 \leq A \leq 10^9$) – количество задач и исходное умение. Далее идут N пар целых чисел a_i, b_i ($1 \leq a_i \leq 10^9$, $1 \leq b_i \leq 10^9$) – соответственно сколько умения нужно для решения i-й задачи и сколько умения прибавится после её решения.</p> <p>Выведите одно число – максимальное количество задач, которое Вася сможет решить.</p>	ipс-1758
Контрольная работа №3		
1	<p>Пароль называется криптостойким, если он включает в себя хотя бы одну строчную латинскую букву, хотя бы одну заглавную латинскую букву и хотя бы одну цифру, при этом его длина должна быть не менее 12 символов.</p> <p>Требуется по данному паролю определить, является ли он криптостойким.</p> <p>Входной файл INPUT.TXT содержит строку, состоящую только из латинских букв и цифр. Длина строки не превосходит 100.</p> <p>В выходной файл OUTPUT.TXT выведите «Yes», если пароль является криптостойким и «No» – в противном случае.</p>	аспр-1151



2	<p>Для того чтобы выходить в Интернет, каждому компьютеру присваивается так называемый IP-адрес. Он состоит из четырех целых чисел в диапазоне от 0 до 255, разделенных точками.</p> <p>По заданной строке требуется определить, является ли она правильным IP-адресом.</p> <p>Входной файл INPUT.TXT содержит строку, состоящую не более, чем из 20 символов. Строка состоит только из цифр и ровно трех точек.</p> <p>В выходной файл OUTPUT.TXT выведите «Good», если строка является правильным IP-адресом, и «Bad» – в противном случае.</p>	астр-1154
3	<p>Разведкой был перехвачен ряд шифровок, которые передавал Джеймс Бонд. Известно, что каждое послание зашифровано методом циклического сдвига. Суть которого в том, что каждая буква заменяется на букву, отстоящую в алфавите от первой на определенном расстоянии. Это расстояние называется знаменателем шифра. Так, при знаменателе шифра 2 буква D превратится в F, буква Q – в S, а Z – в V. Известно, что Бонд использует знаменатели от 0 до 25, и составляет послания исключительно из заглавных букв английского алфавита. Знаменатели в шифровках постоянно меняются, так что расшифровать содержимое послания будет не просто. После тщательного анализа удалось примерно определить предмет посланий. Теперь для каждого послания точно известно одно из входящих туда слов.</p> <p>В первой строке входного файла INPUT.TXT содержится строка с перехваченным посланием, а во второй строке – слово, которое обязательно присутствует в этом послании. Обе строки состоят только из заглавных английских букв и содержат не больше 40 символов.</p> <p>В выходной файл OUTPUT.TXT выведите расшифрованный текст, либо сообщение «IMPOSSIBLE», если разгадать шифровку невозможно. В тех случаях, когда расшифровка возможна с различными знаменателями, то следует вывести вариант с наименьшим таким значением.</p>	астр-295
4	<p>Дана непустая строка S длиной N символов. Будем считать, что элементы строки нумеруются от 1 до N.</p> <p>Для каждой i-й позиции строки S определим подстроку, заканчивающуюся в этой позиции, которая совпадает с некоторым началом всей строки S и имеет длину, меньшую, чем i (т.е. не равна i-му префиксу исходной строки). Значением префикс-функции P(i) будем считать длину этой подстроки.</p> <p>Требуется для всех i от 1 до N вычислить значение P(i).</p> <p>В единственной строке входного файла INPUT.TXT записана строка, состоящая из символов с кодами ASCII от 33 до 127. Длина строки не превышает 10^6.</p> <p>В выходной файл OUTPUT.TXT выведите все значения префикс-функции.</p>	астр-1160
5	<p>Дана непустая строка S длиной N символов. Будем считать, что элементы строки нумеруются от 1 до N.</p> <p>Для каждой i-й позиции строки S определим Z-блок как наибольшую подстроку, которая начинается в этой позиции и совпадает с некоторым началом всей строки S. Значением Z-функции Z(i) будем считать длину этого Z-блока. При $i=1$ будем считать, что $Z(1)=0$, несмотря на то, что в начале строки строка совпадает сама с собой.</p>	астр-1161



	<p>Требуется для всех i от 1 до N вычислить значение $Z(i)$. В единственной строке входного файла INPUT.TXT записана строка, состоящая из символов с кодами ASCII от 33 до 127. Длина строки не превышает 10^6. В выходной файл OUTPUT.TXT выведите все значения Z-функции.</p>	
Контрольная работа №4		
1	<p>После того как Робинзон Крузо построил свою лодку, он решил обследовать близлежащие острова архипелага из N островов. Целью своего путешествия он выбрал N-й остров, надеясь, что обнаружит на нем туземное поселение и британского консула. Опасаясь внезапных тропических штормов, он решил не отплывать далеко от берега и переплывать с одного острова на другой, только если расстояние между берегами этих островов (и, следовательно, плавание вдали от берега) не превышает D километров. Для упрощения задачи все острова будем считать кругами с разным радиусом. Напишите программу, которая поможет Робинзону определить, сможет ли он добраться до N-го острова. Во входном файле в первой строке содержатся два целых числа N ($2 \leq N \leq 100$) и D ($5 \leq D \leq 50$), разделенных одним пробелом – число островов и максимальное расстояние, на которое отваживается совершить плавание Робинзон. В следующих N строках находится по три целых числа X_i, Y_i, R_i ($0 \leq X_i, Y_i \leq 1000, 1 \leq R_i \leq 50, 1 \leq i \leq N$) через один пробел – координаты и радиус острова. Первый остров является стартовым, а N-й – целью путешествия. В выходной файл вывести "YES", если Робинзон сумеет добраться до цели, или "NO", если цели невозможно достичь для заданного ограничения D.</p>	ipс-671
2	<p>Числовой лабиринт представляет собой квадрат размером $N \times N$ клеток, заполненный целыми числами в диапазоне от 1 до 9. Число в клетке означает, на сколько клеток нужно пройти по прямой за день из этой клетки в любом из восьми направлений по горизонтали, вертикали или диагонали. За пределы квадрата выходить нельзя. Из Клондайка в левом верхнем углу квадрата нужно попасть на опушку леса – на любую клетку нижнего или правого края квадрата. Во входном файле в первой строке находится целое число N ($2 \leq N \leq 30$), далее следует N строк по N целых чисел в строке через один пробел. В выходной файл вывести минимальное число дней для прохождения лабиринта. Если выйти из лабиринта невозможно, то вывести число 0.</p>	ipс-117
3	<p>Король Флатландии решил направить свою страну по пути научно-технического прогресса и повелел построить K железных дорог таким образом, чтобы из любого города страны можно было проехать до любого другого по железной дороге прямо или с пересадками через другие города. Железнодорожные компании, заинтересованные в финансировании, предложили министру транспорта откат в случае принятия их предложений. Хитрый министр решил построить дороги таким образом, чтобы не только обеспечить выполнение приказа короля, но и получить максимальный доход. В первой строке ввода содержатся три целых числа, разделенных пробелами – количество городов в стране N ($1 < N \leq 30000$), число предложений от компаний M ($N-1 \leq M \leq 10^5$) и количество дорог K ($N-1 \leq K \leq M$). Далее</p>	ipс-820



	<p>следует M строк, содержащих по три целых числа – номера городов a_i и b_i ($1 \leq a_i < b_i \leq N$), между которыми предлагается построить дорогу, и величина отката c_i ($0 \leq c_i \leq 10^9$). Пары городов в списке предложений не повторяются.</p> <p>Вывести одно целое число – максимальный доход министра. Если выполнить приказ короля невозможно, вывести число -1.</p>	
4	<p>Заяц решил построить себе новую избушку на одной из полян в лесу. Всего в лесу N полян, связанных между собой K тропинками разной длины. Между двумя полянами проложено не более одной тропинки. Тропинки не пересекаются между собой, так как прокладывались с использованием переходных мостов и подземных переходов. На полянах A и B уже построили себе домики Волк и Лиса, а Зайцу, естественно, хочется жить как можно дальше от них. Требуется выбрать полянку для строительства избушки, чтобы величина $1a+1b$ была минимальной, где a – наименьшее расстояние по тропинкам от домика Зайца до домика Волка, а b – наименьшее расстояние от домика Зайца до домика Лисы. Если нет пути между полянами, то соответствующее слагаемое принимаем равным 0.</p> <p>Ввод</p> <p>Во входном файле в первой строке четыре целых числа через один пробел: N, K, A и B – количество полян и тропинок, номера полян с домиками Волка и Лисы ($3 \leq N \leq 20, 1 \leq K \leq 100, 1 \leq A \leq N, 1 \leq B \leq N$). Далее следует K строк с информацией о тропинках. В каждой строке содержится три целых числа через пробел: номера двух полянок, связанных тропинкой, и длина тропинки (от 1 до 1000).</p> <p>Вывод</p> <p>В выходной файл вывести все полянки с минимальным значением указанного выражения в порядке возрастания номеров. Каждый номер поляны выводить на отдельной строке.</p>	ірс-119
5	<p>В стране N городов, некоторые из которых соединены между собой дорогами. Для того, чтобы проехать по одной дороге требуется один бак бензина. В каждом городе бак бензина имеет разную стоимость. Вам требуется добраться из первого города в N-ый, потратив как можно меньшее количество денег.</p> <p>Во входном файле INPUT.TXT записано сначала число N ($1 \leq N \leq 100$), затем идет N чисел, i-ое из которых задает стоимость бензина в i-ом городе (все числа целые из диапазона от 0 до 100). Далее идет число M - количество дорог в стране, далее идет описание самих дорог. Каждая дорога задается двумя числами - номерами городов, которые она соединяет. Все дороги двухсторонние (то есть по ним можно ездить как в одну, так и в другую сторону); между двумя городами всегда существует не более одной дороги; не существует дорог, ведущих из города в себя.</p> <p>В выходной файл OUTPUT.TXT выведите одно число - суммарную стоимость маршрута или -1, если добраться невозможно.</p>	астр-133

3.2.2. Примеры вопросов теста

№ п/п	Категория	Текст вопроса	Правильный ответ
1	Синтаксис	Идентификатором является совокупность – любых символов	набор от 1 до 32 латинских букв, арабских цифр и



		<ul style="list-style-type: none">– любых символов, имеющих двоичный код– набор от 1 до 32 латинских букв, арабских цифр и знака подчеркивания, который не начинается с цифры– любое сочетание русских букв	знака подчеркивания, который не начинается с цифры
2	Типы данных	Из приведенных ниже переменных базовых типов наибольшую длину имеет переменная типа <ul style="list-style-type: none">- short- char- double- long	double
3	Константы	Целая константа задается <ul style="list-style-type: none">- последовательностью цифр без наличия других знаков- последовательностью цифр и точки- последовательностью букв- с помощью спецификатора int	с помощью спецификатора int
4	Переменные	Какова максимальная длина переменной целого типа в байтах?	8
5	Операции	Какая из бинарных операций, которые приводятся ниже, приоритетнейшая? <ul style="list-style-type: none">- %- =- +=- *=	%
6	Ввод-вывод	С помощью какого оператора получена эта строка на экране A = 27.457 <ul style="list-style-type: none">- printf (A :3);- printf ("%d", A);- printf ("A = %f", A);- printf ("A = %6.3f", A);- printf ("A = %7.5f", A);	printf ("A = %6.3f", A);
7	Логические выражения	Какой записи соответствует условие "значение A находится вне отрезка [0,10]"? <ul style="list-style-type: none">- A < 0 && A > 10- A > 0 && A < 10- A > 0 A < 10- A < 0 A > 10- A > 0 && A > 10	A < 0 A > 10
8	Выражения	Если n-целое, то целую часть отношения 100: n можно записать так ... <ul style="list-style-type: none">- 100 / n- 100. / n	100 / n



		<ul style="list-style-type: none">- $100 / (n + 0.0)$- $100 / . n$	
9	Условия	Записать выражение, которое присваивает z значение 1, если точка $(x; y)$ принадлежит внутренности круга радиуса 1 с центром в точке $(1; 0)$ и 0 в противоположном случае. <ul style="list-style-type: none">- <code>if (x<1, y<1) z=1; else z=0;</code>- <code>if (x<1 && y<1) z=1; else z=0;</code>- <code>if(x*x+y*y<1) z=1; else z=0;</code>- <code>if((x-1)*(x-1)+y*y<1) z=1; else z=0;</code>	<code>if ((x-1)*(x-1)+y*y<1) z=1; else z=0;</code>
10	Циклы	В операторе <code>for (E1; E2; E3) s;</code> можно опускать точку с запятой <ul style="list-style-type: none">- после E1- после E2- после s- нигде	нигде
11	Функции	Если функция типизированная, то выход из нее осуществляется <ul style="list-style-type: none">- оператором <code>return</code>- оператором <code>return e</code>- по достижению конца- может иметь место другой выход	оператором <code>return e</code>
12	Указатели	Отметьте вариант, где указатель используется неправильно <ul style="list-style-type: none">- <code>int k, *m; scanf("%d",&k); *m=k; cout<<*m%2;</code>- <code>int k, *m; scanf("%d",m); k=*m; cout<<k%2;</code>- <code>int k=2, *m; m=new int; cin>>m; cout<<m%k;</code>- <code>int k=2, *m; m=new int; cin>>*m; cout<<*m%k;</code>	<code>int k=2, *m; m=new int; cin>>m; cout<<m%k;</code>
13	Выделение памяти	Если локальный массив нужно расположить в свободном поле памяти, то это можно сделать <ul style="list-style-type: none">- вне функций- среди описаний формальных параметров- внутри функции- иное	вне функций
14	STL	Экземпляр класса <code>map</code> хранится в памяти как <ul style="list-style-type: none">- множество пар ключ-значение- хэш-таблица пар ключ-значение- пара множеств ключей и значений- пара указателей ключ-значение	множество пар ключ-значение

3.2.3 Вопросы к зачету

1. Структура программы (на примере C++).
2. Основные виды лексем универсальных ЯП (на примере C+).
3. Выражения языка программирования (на примере C++).
4. Идентификаторы, области видимости деклараций (на примере C++).
5. Ввод-вывод в C++ и в C.



6. Основные операторы универсальных ЯП (на примере C++).
7. Понятие типа данных, классификация типов (на примере C++).
8. Простые типы данных языка C++.
9. Строки в C++.
10. Массивы в C++.
11. Структуры в C++.
12. Функции. Объявление, описание, вызов.
13. Механизмы передачи данных из/в функции: параметры и возвращаемое значение. Способы передачи параметров.
14. Потоки и файлы в C++. Схема обращения к файлам.

Вопросы для зачета 1 семестр:

1. Идентификаторы. Резервированные слова.
2. Объявление и инициализация переменных.
3. Базовые типы данных.
4. Специальные символы и константы.
5. Модификатор const. Назначение, использование при объявлении в C++ и параметров функций.
6. Арифметические операции и присваивание.
7. Математические функции.
8. Сравнение и логические операции.
9. Поразрядные операции.
10. Приоритеты операций.
11. Функции для вывода и ввода в языке C. Спецификаторы формата.
12. Ввод и вывод в C++.
13. Форматированный ввод и вывод в C++.
14. Последовательности операторов и блоки. Области действия имен.
15. Операторы ветвления.
16. Операторы цикла. goto и другие операторы.
17. Структурное программирование.
18. Массивы. Доступ к элементам.
19. Строки в C. Функции для работы со строками
20. Структуры. Псевдонимы типов. Инициализация структур. Доступ к полям.
21. Объединения.
22. Использование vector из C++.
23. Использование string из C++.
24. Определение функции. Вызов функции. Заголовок и тело функции. Объявление функции (прототип).
25. Параметры и возвращаемое значение. Оператор return.
26. Функции с переменным количеством параметров.
27. Параметры по умолчанию в C++.
28. Рекурсия.
29. Модули и компиляция программы.
30. Заголовочные файлы.
31. Препроцессор.
32. Объявления extern, static и inline.
33. Указатели. Передача по указателю, операции взятия адреса и разыменования.



Вопросы для экзамена 2 семестр:

1. Архитектура компьютера, её влияние на технологии разработки программ, языки программирования.
2. Кроссплатформенность, способы её обеспечения.
3. Характеристики центрального процессора: разрядность, тактовая частота, ядра,
4. Устройство центрального процессора: регистры, УУ, АЛУ, адресная шина, кэш
5. Система команд.
6. Память (доступ, адресация, управление памятью в ОС, разделение памяти).
7. Представление целых чисел в памяти.
8. Представление вещественных чисел в памяти.
9. Указатели. Внутреннее представление в памяти.
10. Назначение стека, основные операции. Аппаратный стек.
11. Назначение очереди, основные операции. Очередь сообщений.
12. Списки односвязные и двусвязные. Отличия, преимущества.
13. Устройства отображения информации (видеокарта и встроенная в ЦП)
14. Графика (растровая и векторная)
15. Консоль (исторически и виртуальная).
16. Библиотеки для работы с графикой
17. Библиотеки для работы с консолью.
18. Внешняя память: виды, физическая и логическая организация
19. Внешняя память: файлы/папки.
20. Открытие и закрытие файлов. Ввод и вывод в файл.
21. Работа с бинарными файлами.
22. Таймер, определение текущего времени и измерение интервалов времени.
23. Потоки управления.
24. Распараллеливание программ.
25. Атомарные переменные.

3.2.4 Экзаменационные билеты

№ п/п	Вопросы
1	1. Архитектура фон Неймана. Сегментация программы. Исполнительный цикл процессора. 2. Строки. Массивы char и контейнеры string. 3. Задача.
2	1. Парадигмы программирования. Классификации языков программирования. 2. Функции. Объявление, описание, вызов. 3. Задача.
3	1. Состав языка программирования: алфавит, лексемы, синтаксис, семантика. 2. Механизмы передачи данных из/в функции: параметры и возвращаемое значение. Способы передачи параметров. 3. Задача.
4	1. Системы программирования: определение, состав, схема работы. 2. Структуры. 3. Задача.
5	1. Технологический цикл разработки программ.



	2. Объединения. 3. Задача.
6	1. Классификация программных ошибок. 2. Указатели. Динамическое выделение и освобождение памяти. 3. Задача.
7	1. Структура программы (на примере C++). 2. Двумерные массивы. 3. Задача.
8	1. Основные виды лексем универсальных ЯП (на примере C++). 2. Файлы. Схема обращения программ к файлам. 3. Задача.
9	1. Выражения языка программирования. 2. Строковый ввод и вывод. 3. Задача.
10	1. Идентификаторы, области видимости деклараций. 2. Адаптер stack. 3. Задача.
11	1. Основные операторы универсальных ЯП. 2. Контейнеры vector, list, queue, deque. Итераторы. 3. Задача.
12	1. Понятие типа данных, классификация типов. 2. Контейнеры set, multiset, unordered_set. 3. Задача.
13	1. Ввод и вывод в C++ через потоки. 2. Контейнеры pair, map, multimap, unordered_map. 3. Задача.
14	1. Функции форматного ввода-вывода. 2. Адаптер bitset. 3. Задача.
15	1. Циклы с предусловием и с постусловием. 2. Базовые функции и алгоритмы работы с контейнерами. 3. Задача.
16	1. Циклы for. 2. Лямбда-функции. 3. Задача.
17	1. Массивы и контейнеры array. 2. Переопределение операторов. 3. Задача.

3.2.5 Задачи к экзамену

Примеры простых задач к экзамену

1) Исключить из строки группы символов, расположенные между скобками (.). Сами скобки тоже должны быть исключены. Предполагается, что внутри каждой пары скобок нет других скобок.

Ввод: в первой строке содержится последовательность символов, оканчивающаяся символом перехода на новую строку. Длина последовательности не превышает 256.



Вывод: вывести измененную строку.

Пример ввода: Text1 (text2) text3. Пример вывода: Text1 text3.

2) Даны натуральные числа a, b ($1 \leq a \leq b \leq 1000$). Получить количество всех простых чисел, входящих в диапазон $[a; b]$

Пример ввода: 4 15 Пример вывода: 4.

Примеры задач средней сложности к экзамену

3) Найти количество всех прямоугольных параллелепипедов, объём которых равен заданному натуральному числу v ($1 \leq v \leq 100$) и стороны которых выражены натуральными числами. При этом решения, которые получаются перестановкой размеров рёбер параллелепипеда считать разными.

Ввод: В первой строке ввода содержится число v .

Вывод: Количество всех прямоугольных параллелепипедов.

Пример ввода: 9 Пример вывода: 6.

4) Необходимо представить целое число N в виде суммы M примерно равных целых чисел. Будем считать, что числа примерно равны, если они отличаются друг от друга не более чем на единицу.

Во входном файле INPUT.TXT записаны два натуральных числа N и M через пробел, каждое из которых не превосходит 30000.

Выходной файл OUTPUT.TXT должен содержать M примерно равных целых чисел, сумма которых должна быть равна N . Все числа следует вывести в одной строке в порядке неубывания через пробел.

Пример ввода: 13 4 Пример вывода: 3 3 3 4.



4. Порядок проведения и критерии оценивания промежуточной аттестации

4.1. Порядок проведения промежуточной аттестации

На зачете в 1 и 2 семестре студенту будет предложен вопрос по одному из разделов курса, при ответе на который обучающийся должен продемонстрировать понимание вопроса и проиллюстрировать его разбором практического примера. В компьютерном классе возможна сдача зачёта в виде теста.

На экзамене во 2 семестре студенту будет предложен компьютерный тест из 20 вопросов и экзаменационный билет, который содержит 3 вопроса из разных разделов дисциплины – два теоретических и один практический в виде решения задачи.

Экзамен проводится в очной форме по экзаменационным билетам. Процедура прохождения экзамена не является обязательной, если по результатам текущего контроля БРС у студента положительная оценка и он с ней согласен. В каждом билете один теоретический вопрос и одна задача. Экзамен принимается в устной форме. Студент должен находиться в аудитории на протяжении всей процедуры экзамена. На подготовку к ответу студенту отводится не более 30 мин. Когда обучающийся будет готов к ответу, ему задаются контрольные вопросы по содержанию билета. Студент должен устно ответить на эти вопросы в течение 5 мин. На этом основании преподаватель выставляет баллы за экзаменационную работу.

4.2. Критерии оценивания промежуточной аттестации по видам оценочных средств

4.2.1 Критерии оценивания контрольной работы

Контрольные работы представляют собой наборы задач по программированию на изучаемые темы, которые студентам следует в заданные сроки в соответствующих семестрах решать во внеучебное время и сдавать на специализированных веб-сайтах с автоматической проверкой на наборах тестов (<http://acmp.ru>, <http://ipc.susu.ac.ru>).

За полное решение каждой задачи начисляется 2 балла, за частичное – 1 балл, и так до 10 баллов за каждую из 4 контрольных работ, всего до 40 баллов за 2 семестра.

4.2.2 Критерии оценивания теста

«зачтено» – тест выполнен с результатом не менее 50% (10-20 баллов);

«не зачтено» – тест выполнен с результатом менее 50% (0-9 баллов).

4.2.3 Критерии оценивания ответа на экзамене

На экзамене студенту будет предложен компьютерный тест из 20 вопросов.

Оценка «неудовлетворительно» выставляется при числе правильных ответов на вопросы теста менее 10 (баллов).

Правильные ответы на 10 и более вопросов позволяют студенту получить оценку «удовлетворительно» и закончить экзамен или – при желании получить более высокую оценку – взять билет, при ответе на вопросы которого экзаменуемый должен продемонстрировать знание теоретических понятий темы вопроса и проиллюстрировать их разбором практического примера.

Экзаменационный билет содержит 3 вопроса из разных разделов дисциплины – 2 теоретических (до 10 баллов за ответ на каждый) и 1 практический в виде решения задачи на сайте с автоматической проверкой на наборах тестов (<http://acmp.ru>, <http://ipc.susu.ac.ru>) (до 10 баллов), ещё до 10 баллов могут быть добавлены за решение дополнительной задачи – всего до 40 баллов.

При этом для получения оценки «отлично» необходимо набрать при ответе на вопросы билета не менее 40 баллов, а для получения оценки «хорошо» - не менее 30 баллов.



Также могут быть оценены дополнительными баллами (до 40 баллов) успешные решения задач на студенческих олимпиадах по программированию.

В соответствии с Положением о балльно-рейтинговой системе (БРС) оценка может быть выставлена на основе суммы баллов текущей аттестации за семестр и баллов за экзамен.

4.2.4 Критерии оценивания теста

Курсовая работа оценивается следующим образом:

Анализ предметной области - 3 балла

Постановка задачи - 7 баллов

Формализация задачи - 10 баллов

Схемы алгоритмов - 15 баллов

Реализация программы - 15 баллов

Тестирование и исправление ошибок в программе - 15 баллов

Оформление пояснительной записки - 10 баллов

Защита курсовой работы - 25 баллов

В сумме: 100 баллов.

4.3. Результаты промежуточной аттестации и уровни сформированности компетенций

Зачет в 1 и 2 семестре выставляется на основе балльно-рейтинговой системы по итогам текущего контроля выполнения контрольных работ. В компьютерном классе возможна сдача зачёта в виде теста.

Возможные оценки:

«зачтено» – выполнил все работы и усвоил материал;

«не зачтено» – не все работы выполнил, материал не усвоил.

Экзаменационная оценка во 2 семестре выставляется на основе балльно-рейтинговой системы по итогам текущего контроля выполнения контрольных работ, выполнения экзаменационного теста и ответа на билет.

Контрольные работы – 0-40 б.

Тест – 0-20 б.

Экзамен – 0-40 б.

Всего: 100б.

Возможные оценки на экзамене:

«Отлично» («5») – студент глубоко и полно владеет содержанием учебного материала и понятийным аппаратом; умеет связывать теорию с практикой, иллюстрировать примерами, фактами, данными научных исследований; обозначает межпредметные связи. Делает выводы логично, четко. Ясно и кратко излагает ответы на поставленные вопросы; умеет обосновывать свои суждения и профессионально-личностную позицию по излагаемому вопросу. Ответ носит самостоятельный характер.

«Хорошо» («4») – ответ студента соответствует указанным выше критериям, но содержание ответа имеет отдельные неточности (несущественные ошибки) в изложении теоретического и практического материала, отличается меньшей обстоятельностью, глубиной и полнотой; допущенные ошибки исправляются студентом после дополнительных вопросов экзаменатора.

«Удовлетворительно» («3») – студент обнаруживает знание и понимание основных положений учебного материала, но излагает его неполно, непоследовательно, допускает неточности



и существенные ошибки в определении понятий, формулировке положений, не привлекает для аргументации ответа основные положения исследовательских, концептуальных и нормативных документов, не умеет обосновать свои суждения; наблюдается нарушение логики изложения. Ответ отличается низким уровнем самостоятельности, не содержит собственной профессионально-личностной позиции.

«Неудовлетворительно» («2») – студент имеет разрозненные, бессистемные знания: не умеет выделять главное и второстепенное; допускает ошибки в определении понятий, формулировке теоретических положений, искажающие их смысл; не ориентируется в нормативно-концептуальных, программно-методических, исследовательских материалах, беспорядочно и неуверенно излагает материал; не умеет соединять теоретические положения с педагогической практикой; не умеет применять знания для обоснования и объяснения фактов.

Особенности проведения процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья обозначены в рабочей программе дисциплины (модуля).

Уровни сформированности компетенций определяется следующим образом:

1. Продвинутый уровень сформированности компетенций соответствует оценке «зачтено»/«отлично»:

Обучающийся владеет знаниями предмета в полном объеме учебной программы, достаточно глубоко осмысливает дисциплину; самостоятельно, в логической последовательности и исчерпывающе отвечает на все вопросы, подчеркивает при этом самое существенное, умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделять в нем главное: устанавливать причинно-следственные связи; четко формирует ответы.

2. Базовый уровень соответствует оценке «зачтено»/«хорошо»:

Обучающийся владеет знаниями дисциплины почти в полном объеме программы (имеются пробелы знаний только в некоторых, особенно сложных разделах); самостоятельно и отчасти при наводящих вопросах дает полноценные ответы на вопросы; не всегда выделяет наиболее существенное, не допускает вместе с тем серьезных ошибок в ответах.

3. Пороговый уровень соответствует оценке «зачтено»/«удовлетворительно»:

Обучающийся владеет основным объемом знаний по дисциплине; проявляет затруднения в самостоятельных ответах, оперирует неточными формулировками; в процессе ответов допускает ошибки по существу вопросов.

4. Низкий уровень соответствует оценке «не зачтено»/«неудовлетворительно»:

Обучающийся не освоил обязательного минимума знаний предмета, не способен ответить на вопросы даже при дополнительных наводящих вопросах экзаменатора.

