

Управление культуры Администрации города Екатеринбурга  
Муниципальное бюджетное образовательное учреждение  
высшего образования  
«Екатеринбургская академия современного искусства»  
(институт)



УТВЕРЖДАЮ

Ректор

«01» апреля 2024 г.

И.А. Ахьямова

ПРОГРАММА  
вступительного испытания по дисциплине  
**ИНФОРМАТИКА**  
для поступающих на базе среднего общего образования

Направление подготовки  
**09.03.03 Прикладная информатика**

Екатеринбург  
2024

Программа вступительного испытания для поступающих в МБОУ ВО ЕАСИ на базе среднего общего образования по дисциплине «Информатика». Направление подготовки бакалавров 09.03.03 Прикладная информатика

Составители: к.п.н., доцент, директор регионального центра цифровой трансформации образования ГАОУ ДПО СО «Институт развития образования»  
Е.Б. Стариченко

## СОДЕРЖАНИЕ

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА.....	4
СТРУКТУРА ВСТУПИТЕЛЬНОГО ИСПЫТАНИЯ.....	4
КРИТЕРИИ ОЦЕНИВАНИЯ .....	4
СОДЕРЖАНИЕ РАЗДЕЛОВ ПРОГРАММЫ.....	8
СПИСОК ЛИТЕРАТУРЫ, РЕКОМЕНДУЕМОЙ ДЛЯ ПОДГОТОВКИ	<b>Ошибка!</b>

**Закладка не определена.**

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Программа предназначена для абитуриентов, поступающих на базе среднего общего образования по направлению подготовки бакалавров 09.03.03 Прикладная информатика, на основании результатов вступительных испытаний, проводимых вузом.

Категории поступающих имеющих право поступать на обучение по результатам общеобразовательных вступительных испытаний, проводимых институтом, самостоятельно определяются в соответствии с Правилами приема граждан на обучение по программам бакалавриата в 2024 году в МБОУ ВО «Екатеринбургская академия современного искусства» (институт) (далее Правила приема).

Максимальная продолжительность вступительного испытания 3 часа 55 минут (235 мин), допускается досрочное завершение тестирования по желанию испытуемого.

Объявление результата вступительного испытания, ознакомление с экзаменационной работой, прием и рассмотрение апелляций производится в соответствии с Правилами приема.

### СТРУКТУРА ВСТУПИТЕЛЬНОГО ИСПЫТАНИЯ

Вступительное испытание содержит задания по 10 темам курса «Информатика и информационно-коммуникационные технологии», составляющим основное содержание предмета.

Вступительное испытание состоит из 2 частей, включающих 27 заданий. На выполнение части 1 работы рекомендуется отводить 1,5 часа (90 минут). На выполнение заданий части 2-2,5 часа (150 минут).

Часть 1 содержит 23 задания с кратким ответом.

Часть 2 содержит 4 задания с развёрнутым ответом.

### КРИТЕРИИ ОЦЕНИВАНИЯ

№	Название раздела	Макс. балл
1	Цифровая грамотность	25
2	Теоретические основы информатики	25
3	Информационные технологии	25
4	Алгоритмы и программирование	25
	<b>Итого:</b>	<b>100</b>

**Итоговой оценкой за вступительное испытание** является сумма баллов за все задания.

Максимальное количество баллов, которое может получить абитуриент по итогам вступительного испытания, равняется 100 баллам. Минимальное количество баллов,

подтверждающее положительное прохождение вступительного испытания по Информатике и информационно-коммуникационным технологиям составляет 42 балла.

Согласно Приказу Министерства науки и высшего образования Российской Федерации от 1 марта 2023 г. № 231 «Об утверждении особенностей приема на обучение в организации, осуществляющие образовательную деятельность, по программам бакалавриата, программам специалитета, программам магистратуры и программам подготовки научно-педагогических кадров в аспирантуре (адъюнктуре)», предусмотренных частями 7 и 8 статьи 5 федерального закона от 17 февраля 2023 г. № 19-ФЗ «Об особенностях правового регулирования отношений в сферах образования и науки в связи с принятием в Российскую Федерацию Донецкой народной республики, Луганской народной республики, Запорожской области, Херсонской области и образованием в составе Российской Федерации новых субъектов – Донецкой народной республики, Луганской народной республики, Запорожской области, Херсонской области и о внесении изменений в отдельные законодательные акты Российской Федерации», лица, поступающие в МБОУ ВО ЕАСИ из выше перечисленных республик, имеют право по желанию выбирать вступительные испытания на обучение в форме собеседования.

Собеседование осуществляется в форме устного опроса по экзаменационным билетам. Экзаменационный билет состоит из двух теоретических вопросов и двух задач по программированию или практической обработке информации с использованием программного обеспечения. Билет включает вопросы и задания разного уровня сложности. Степень сложности и трудоемкости содержания билетов одинакова.

### **Вопросы для подготовки к экзамену**

1. Принципы работы компьютера. Типы компьютеров.
2. Основные компоненты компьютера и их назначение.
3. Программное обеспечение компьютера.
4. Операционная система.
5. Файловая система.
6. Лицензирование программного обеспечения и цифровых ресурсов.
7. Компьютерные сети.
8. Глобальная сеть Интернет. Адресация в сети Интернет. Система доменных имён.
9. Сервисы Интернета.
10. Защита информации в Интернете.
11. Информация, данные и знания. Дискретность данных. Информационные процессы.
12. Кодирование информации. Алфавит. Дискретное представление информации.
13. Измерение информации.
14. Передача информации.
15. Цифровое представление непрерывных данных.
16. Модели и моделирование. Этапы компьютерного моделирования.

17. Графы.
18. Деревья.
19. Системы счисления. Перевод чисел из одной системы счисления в другую.
20. Алгебра логики. Логические операции. Логические функции. Логические элементы компьютера.
21. Компьютерные интеллектуальные системы.
22. Алгоритмы. Алгоритмические конструкции. Этапы решения задач на компьютере.
23. Основные конструкции языка программирования.

### **Критерии оценивания устного ответа**

Вступительное испытание по информатике в форме собеседования представляет устный ответ на вопросы экзаменационного билета и объяснение хода решения задач и обоснование полученных результатов.

Ответ на каждое задание билета оценивается от 0 до 25 баллов:

25 баллов – по своему содержанию устный ответ полностью соответствует вопросу, содержит все необходимые теоретические факты и обоснованные выводы, а его изложение логически грамотно, отличается последовательностью и аккуратностью; при решении задачи по программированию правильно выбран способ решения, само решение сопровождается необходимыми объяснениями, верно выполнен алгоритм решения, решение записано последовательно, аккуратно и синтаксически верно по правилам какого-либо языка или системы программирования; при решении задачи по практической обработке информации самостоятельно или с незначительной помощью выполнены этапы обработки информации и получен верный ответ или иное требуемое представление результата.

15-24 баллов – в изложении устного ответа допущены небольшие пробелы, не исказившие логического и информационного содержания ответа, нет определённой логической последовательности, неточно используется специализированная терминология и символика, допущены один-два недочёта при освещении основного содержания ответа, исправленные по замечанию экзаменатора, допущены ошибка или более двух недочётов при освещении второстепенных вопросов или в выкладках, легко исправленные по замечанию или вопросу экзаменатора. Практические задания выполнены полностью или не менее чем на 80% от объёма задания, но в них имеются недочёты и несущественные ошибки, например, правильно записаны алгоритмы, но в синтаксисе имеются ошибки, программа работает, но не всех случаях выдаёт правильный результат, средство обработки информации выбрано верно, но использовано не в полной мере или некорректно.

5-14 баллов – неполно или непоследовательно раскрыто содержание материала, но показано общее понимание вопроса, имелись затруднения или допущены ошибки в определении понятий, использовании терминологии, чертежах, блок-схемах и выкладках, исправленные после нескольких наводящих вопросов экзаменатора, абитуриент не справился с применением теории в новой ситуации при выполнении

практического задания, но выполнил задания обязательного уровня сложности по данной теме, при знании теоретического материала выявлена недостаточная сформированность основных компетенций. При решении практических заданий работа выполнена в основном верно (объем выполненной части составляет не менее 2/3 от общего объема), но допущены более одной ошибки или двух-трех недочетов в выкладках, чертежах блок-схем или программе, но абитуриент владеет основными навыками работы, требуемыми для решения поставленной задачи.

1-4 баллов – тема в основном не, абитуриент показывает незнание основных понятий, непонимание закономерностей и взаимосвязей, допущены существенные ошибки, показавшие, что абитуриент не владеет обязательными знаниями по данной теме в полной мере, начальный ход решения верен, полученные промежуточные результаты не обоснованы, решение не завершено;

0 баллов – не владеет основными знаниями и умениями в соответствии с требованиями программы вступительного испытания.

Максимальная сумма баллов – 100.

### *Пример экзаменационного билета*

**Вопрос 1.** Операционная система. Понятие, история, назначение, драйверы, утилиты.

**Вопрос 2.** Основные конструкции языка программирования. Назначение, примеры, синтаксис на основе любого выбранного языка.

**Задача 1.** Определите, что будет напечатано в результате работы следующего фрагмента программы.

Бейсик	Алгоритмический язык	Python
<pre>DIM S, K AS INTEGER S = 0 K = 1 WHILE S &lt; 11 S = S + K K = K + 1 WEND PRINT S</pre>	<pre>алг нач цел k, s s := 0 k := 1 нц пока k &lt; 11 s := s + k k := k + 1 кц ВЫВОД s КО</pre>	<pre>s=0 k=1 while k &lt; 11: s=s+k k=k+1 print(s)</pre>
C++		Паскаль
<pre>#include &lt;iostream&gt; using namespace std; int main() { int k, s; s = 0; k = 1; while (k &lt; 11) { s = s + k; k = k + 1; } cout &lt;&lt; s; return 0;</pre>		<pre>var k, s: integer; begin s:=0; k:=1; while k &lt; 11 do begin s:=s+k; k:=k+1; end; write(s); end.</pre>

}	
---	--

**Задача 2.** Рисунок размером 512 на 256 пикселей занимает в памяти 64 Кбайт (без учёта сжатия). Найдите максимально возможное количество цветов в палитре изображения.

## СОДЕРЖАНИЕ РАЗДЕЛОВ ПРОГРАММЫ

### Раздел 1. Цифровая грамотность

Принципы работы компьютера. Персональный компьютер. Выбор конфигурации компьютера в зависимости от решаемых задач. Основные тенденции развития компьютерных технологий. Параллельные вычисления. Многопроцессорные системы. Суперкомпьютеры. Микроконтроллеры. Роботизированные производства. Требования техники безопасности и гигиены при работе с компьютерами и другими компонентами цифрового окружения.

Программное обеспечение компьютеров. Виды программного обеспечения и их назначение. Особенности программного обеспечения мобильных устройств. Операционная система. Понятие о системном администрировании. Установка и деинсталляция программного обеспечения.

Файловая система. Поиск в файловой системе. Организация хранения и обработки данных с использованием интернет-сервисов, облачных технологий и мобильных устройств.

Прикладные компьютерные программы для решения типовых задач по выбранной специализации. Системы автоматизированного проектирования. Лицензирование программного обеспечения и цифровых ресурсов. Проприетарное и свободное программное обеспечение. Коммерческое и некоммерческое использование программного обеспечения и цифровых ресурсов. Ответственность, устанавливаемая законодательством Российской Федерации, за неправомерное использование программного обеспечения и цифровых ресурсов.

Принципы построения и аппаратные компоненты компьютерных сетей. Сетевые протоколы. Сеть Интернет. Адресация в сети Интернет. Система доменных имён. Веб-сайт. Веб-страница. Взаимодействие браузера с веб-сервером. Динамические страницы. Разработка интернет-приложений (сайтов). Сетевое хранение данных. Виды деятельности в сети Интернет.

Сервисы Интернета. Геоинформационные системы. Геолокационные сервисы реального времени (например, локация мобильных телефонов, определение загруженности автомагистралей), интернет-торговля, бронирование билетов, гостиниц. Государственные электронные сервисы и услуги.

Социальные сети – организация коллективного взаимодействия и обмена данными. Сетевой этикет: правила поведения в киберпространстве. Проблема подлинности полученной информации. Открытые образовательные ресурсы.

Техногенные и экономические угрозы, связанные с использованием информационно-коммуникационных технологий. Общие проблемы защиты информации



и информационной безопасности. Средства защиты информации в компьютерах, компьютерных сетях и автоматизированных информационных системах. Правовое обеспечение информационной безопасности. Предотвращение несанкционированного доступа к личной конфиденциальной информации, хранящейся на персональном компьютере, мобильных устройствах. Вредоносное программное обеспечение и способы борьбы с ним. Антивирусные программы. Организация личного архива информации. Резервное копирование. Парольная защита архива.

Информационные технологии и профессиональная деятельность. Информационные ресурсы. Цифровая экономика. Информационная культура.

## **Раздел 2. Теоретические основы информатики**

Информация, данные и знания. Универсальность дискретного представления информации. Двоичное кодирование. Равномерные и неравномерные коды. Условие Фано. Подходы к измерению информации. Сущность объёмного (алфавитного) подхода к измерению информации, определение бита с точки зрения алфавитного подхода, связь между размером алфавита и информационным весом символа (в предположении о равновероятности появления символов), связь между единицами измерения информации: бит, байт, Кбайт, Мбайт, Гбайт. Сущность содержательного (вероятностного) подхода к измерению информации, определение бита с позиции содержания сообщения.

Информационные процессы. Передача информации. Источник, приёмник, канал связи, сигнал, кодирование. Искажение информации при передаче. Скорость передачи данных по каналу связи. Хранение информации, объём памяти. Обработка информации. Виды обработки информации: получение нового содержания, изменение формы представления информации. Поиск информации. Роль информации и информационных процессов в окружающем мире.

Системы. Компоненты системы и их взаимодействие. Системы управления. Управление как информационный процесс. Обратная связь. Системы счисления. Развёрнутая запись целых и дробных чисел в позиционных системах счисления. Свойства позиционной записи числа: количество цифр в записи, признак делимости числа на основание системы счисления. Алгоритм перевода целого числа из  $P$ -ичной системы счисления в десятичную. Алгоритм перевода конечной  $P$ -ичной дроби в десятичную. Алгоритм перевода целого числа из десятичной системы счисления в  $P$ -ичную. Двоичная, восьмеричная и шестнадцатеричная системы счисления, перевод чисел между этими системами. Арифметические операции в позиционных системах счисления. Представление целых и вещественных чисел в памяти компьютера. Кодирование текстов. Кодировка ASCII. Однобайтные кодировки. Стандарт UNICODE. Кодировка UTF-8. Определение информационного объёма текстовых сообщений.

Кодирование изображений. Оценка информационного объёма растрового графического изображения при заданном разрешении и глубине кодирования цвета. Кодирование звука. Оценка информационного объёма звуковых данных при заданных частоте дискретизации и разрядности кодирования.

Алгебра логики. Высказывания. Логические операции. Таблицы истинности логических операций «дизъюнкция», «конъюнкция», «инверсия», «импликация», «эквиваленция». Логические выражения. Вычисление логического значения составного высказывания при известных значениях входящих в него элементарных высказываний. Таблицы истинности логических выражений. Логические операции и операции над множествами. Примеры законов алгебры логики. Эквивалентные преобразования логических выражений. Логические функции. Построение логического выражения с данной таблицей истинности. Логические элементы компьютера. Триггер. Сумматор. Построение схемы на логических элементах по логическому выражению. Запись логического выражения по логической схеме.

Модели и моделирование. Цели моделирования. Соответствие модели моделируемому объекту или процессу. Формализация прикладных задач. Представление результатов моделирования в виде, удобном для восприятия человеком.

Графическое представление данных (схемы, таблицы, графики). Графы. Основные понятия. Виды графов. Решение алгоритмических задач, связанных с анализом графов (построение оптимального пути между вершинами графа, определение количества различных путей между вершинами ориентированного ациклического графа). Деревья. Бинарное дерево. Дискретные игры двух игроков с полной информацией. Построение дерева перебора вариантов, описание стратегии игры в табличной форме. Выигрышные стратегии. Использование графов и деревьев при описании объектов и процессов окружающего мира.

### **Раздел 3. Информационные технологии**

Текстовый процессор. Редактирование и форматирование. Проверка орфографии и грамматики. Средства поиска и автозамены в текстовом процессоре. Использование стилей. Структурированные текстовые документы. Сноски, оглавление. Облачные сервисы. Коллективная работа с документом. Инструменты рецензирования в текстовых процессорах. Деловая переписка. Реферат. Правила цитирования источников и оформления библиографических ссылок. Оформление списка литературы.

Ввод изображений с использованием различных цифровых устройств (цифровых фотоаппаратов и микроскопов, видеокамер, сканеров и других устройств.). Графический редактор. Обработка графических объектов. Растровая и векторная графика. Форматы графических файлов. Обработка изображения и звука с использованием интернет-приложений.

Мультимедиа. Компьютерные презентации. Использование мультимедийных онлайн-сервисов для разработки презентаций проектных работ. Принципы построения и редактирования трёхмерных моделей.

Анализ данных. Основные задачи анализа данных: прогнозирование, классификация, кластеризация, анализ отклонений. Последовательность решения задач анализа данных: сбор первичных данных, очистка и оценка качества данных, выбор и/или построение модели, преобразование данных, визуализация данных, интерпретация результатов. Анализ данных с помощью электронных таблиц. Вычисление суммы, среднего арифметического, наибольшего и наименьшего значений диапазона.

Компьютерно-математические модели. Этапы компьютерно-математического моделирования: постановка задачи, разработка модели, тестирование модели, компьютерный эксперимент, анализ результатов моделирования. Численное решение уравнений с помощью подбора параметра.

Табличные (реляционные) базы данных. Таблица – представление сведений об однотипных объектах. Поле, запись. Ключ таблицы. Работа с готовой базой данных. Заполнение базы данных. Поиск, сортировка и фильтрация записей. Запросы на выборку данных. Запросы с параметрами. Вычисляемые поля в запросах. Многотабличные базы данных. Типы связей между таблицами. Запросы к многотабличным базам данных.

Средства искусственного интеллекта. Сервисы машинного перевода и распознавания устной речи. Идентификация и поиск изображений, распознавание лиц. Самообучающиеся системы. Искусственный интеллект в компьютерных играх. Использование методов искусственного интеллекта в обучающих системах. Использование методов искусственного интеллекта в робототехнике. Интернет вещей. Перспективы развития компьютерных интеллектуальных систем.

#### **Раздел 4. Алгоритмы и программирование**

Определение возможных результатов работы простейших алгоритмов управления исполнителями и вычислительных алгоритмов. Определение исходных данных, при которых алгоритм может дать требуемый результат. Этапы решения задач на компьютере. Язык программирования (Паскаль, Python, Java, C++, C#). Основные конструкции языка программирования.

Типы данных: целочисленные, вещественные, символьные, логические. Ветвления. Составные условия. Циклы с условием. Циклы по переменной. Использование таблиц трассировки. Разработка и программная реализация алгоритмов решения типовых задач базового уровня.

Примеры задач: алгоритмы обработки конечной числовой последовательности (вычисление сумм, произведений, количества элементов с заданными свойствами), алгоритмы анализа записи чисел в позиционной системе счисления, алгоритмы решения задач методом перебора (поиск наибольшего общего делителя двух натуральных чисел, проверка числа на простоту).

Обработка символьных данных. Встроенные функции языка программирования для обработки символьных строк.

Табличные величины (массивы). Алгоритмы работы с элементами массива с однократным просмотром массива: суммирование элементов массива, подсчёт количества (суммы) элементов массива, удовлетворяющих заданному условию, нахождение наибольшего (наименьшего) значения элементов массива, нахождение второго по величине наибольшего (наименьшего) значения, линейный поиск элемента, перестановка элементов массива в обратном порядке. Сортировка одномерного массива. Простые методы сортировки (например, метод пузырька, метод выбора, сортировка вставками). Подпрограммы.

## **СПИСОК ЛИТЕРАТУРЫ, РЕКОМЕНДУЕМОЙ ДЛЯ ПОДГОТОВКИ**

1. *Алексеев В.А.* Информатика. Практические работы. М.: Лань, 2024. 256 с.
2. *Бажукова Е.Н., Заливадный М.С., Бажукова Е.Н.* Музыкальная информатика. М.: Планета музыки, 2023. 208 с.
3. *Баяк О.А.* Математика и информатика для лингвистов. Учебное пособие. М.: Прометей, 2023. 162 с.
4. *Босова Л.Л., Босова А.Ю.* Информатика. Базовый уровень. Учебное пособие для СПО. Часть 1. М.: Просвещение, 2024. 304 с.
5. *Босова А.Ю., Босова Л.Л.* Информатика. Базовый уровень. Учебное пособие для СПО. Часть 2. М.: Просвещение, 2024. 272 с.
6. *Волк В.К.* Информатика. М.: Юрайт, 2024. 227 с.
7. *Волк В.К.* Информатика. Углубленный уровень: 10-11 классы. М.: Юрайт, 2024. 228 с.
8. *Гаврилов М.В., Климов В.А.* Информатика и информационные технологии. М.: Юрайт, 2023. 356 с.
9. *Гилярова М.Г.* Информатика для медицинских колледжей. Учебник. М.: Феникс, 2024. 528 с.
10. *Гомбоева И.С.* Информатика. Сборник профессионально ориентированных задач по профессии «Повар, кондитер». М.: Лань, 2024. 72 с.
11. *Горбенко А.О.* Бизнес-информатика. Введение в профессию. Учебное пособие. М.: КноРус, 2022. 152 с.
12. *Гуриков С.Р.* Информатика. М.: Инфра-М, 2023. 566 с.
13. *Гусева Е.Н.* Информатика. Учебное пособие. М.: Флинта, 2023. 208 с.
14. *Дмитрик Н.А.* Правовая информатика. Учебник. М.: Инфотропик Медиа, 2022. 172 с.
15. *Зараменских Е.П.* Основы бизнес-информатики. М.: Юрайт, 2023. 471 с.
16. *Зубова Е.Д.* Информатика и ИКТ. М.: Лань, 2023. 180 с.
17. Информатика / под ред. В. В. Трофимов. М.: Юрайт, 2023. 796 с.
18. Информатика для гуманитариев / под ред. Г. Е. Кедровой. М.: Юрайт, 2023. 663 с.
19. Информатика и математика / под ред. В. Д. Элькина. М.: Юрайт, 2023. 403 с.
20. *Иопа Н.И.* Информатика (для технических направлений). М.: КноРус, 2022. 470 с.
21. *Логунова О.С.* Информатика. Курс лекций. М.: Инфра-Инженерия, 2022. 164 с.
22. *Лопатин В.М.* Информатика. М.: Лань, 2021. 216 с.
23. *Мойзес О.Е., Кузьменко Е.А.* Информатика. Углубленный курс. М.: Юрайт, 2023. 165 с.
24. *Нагаева И.А.* Арт-информатика. Учебное пособие. М.: Директмедиа Паблишинг, 2022. 384 с.
25. *Новожилков О.П.* Информатика в 2 частях. Часть 1. М.: Юрайт, 2024. 321 с.
26. *Новожилков О.П.* Информатика в 2 частях. Часть 2. М.: Юрайт, 2024. 303 с.

27. *Обмачевская С.Н.* Медицинская информатика. Курс лекций. М.: Лань, 2022. 184 с.
28. *Олифер В.Г.* Компьютерные сети. Принципы, технологии, протоколы. СПб.: Питер, 2024. 1008 с.
29. Правовая информатика / под ред. С. Г. Чубукова. М.: Юрайт, 2024. 315 с.
30. *Родичев Ю.А.* Информационная безопасность. Национальные стандарты Российской Федерации. 3-е изд. Учебное пособие. СПб.: Питер, 2021. 896 с.
31. *Симонович С.В.* Информатика. Базовый курс: Учебник для вузов. СПб.: Питер, 2023. 640 с.
32. *Торадзе Д.Л.* Информатика. М.: Юрайт, 2023. 159 с.
33. *Трофимов В.В., Барабанова М.И.* Информатика в 2 томах. Том 1. М.: Юрайт, 2023. 554 с.
34. *Украинцев Ю.Д.* Информатизация общества. М.: Лань, 2021. 224 с.
35. *Федотова Е.Л.* Информатика. Учебное пособие. М.: Инфра-М, 2024. 453 с.
36. *Фомичёв В.М., Мельников Д.А.* Криптографические методы защиты информации в 2 частях. Часть 1. Математические аспекты. М.: Юрайт, 2024. 210 с.
37. *Фомичёв В.М., Мельников Д.А.* Криптографические методы защиты информации в 2 частях. Часть 2. Системные и прикладные аспекты. М.: Юрайт, 2024. 246 с.
38. *Черпаков И.В.* Теоретические основы информатики. М.: Юрайт, 2024. 354 с.
39. *Чугунов А.В.* Социальная информатика. М.: Юрайт, 2024. 257 с.
40. Экономическая информатика / под ред. В. П. Полякова. М.: Юрайт, 2023. 516 с.
41. *Яценков В.С.* Введение в эволюционную информатику. М.: ДМК Пресс, 2020. 276 с.

## ПРИМЕР ЭКЗАМЕНАЦИОННОГО ЗАДАНИЯ

В экзаменационных заданиях используются следующие соглашения:

1. Обозначения для логических связок (операций):

а) *отрицание* (инверсия, логическое НЕ) обозначается (например,  $\neg A$ );

б) *конъюнкция* (логическое умножение, логическое И) обозначается  $\wedge$  (например,  $A \wedge B$ ) либо  $\&$  (например,  $A \& B$ );

в) *дизъюнкция* (логическое сложение, логическое ИЛИ) обозначается  $\vee$  (например,  $A \vee B$ ) либо  $|$  (например,  $A | B$ );

г) *следование* (импликация) обозначается  $\rightarrow$  (например,  $A \rightarrow B$ );

д) символ 1 используется для обозначения истины (истинного высказывания); символ 0 - для обозначения лжи (ложного высказывания).

2. Два логических выражения, содержащих переменные, называются *равносильными* (эквивалентными), если значения этих выражений совпадают при любых значениях переменных. Так, выражения  $A \rightarrow B$  и  $(\neg A) \vee B$  равносильны, а  $A \vee B$  и  $A \wedge B$  - нет (значения выражений разные, например, при  $A=1, B=0$ ).

3. Приоритеты логических операций: инверсия (отрицание), конъюнкция (логическое умножение), дизъюнкция (логическое сложение), импликация (следование), эквивалентность (равносильность). Таким образом,  $\neg A \wedge B \vee C \wedge D$  совпадает с  $((\neg A) \wedge B) \vee (C \wedge D)$ . Возможна запись  $A \wedge B \wedge C$  вместо  $(A \wedge B) \wedge C$ . То же относится и к дизъюнкции: возможна запись  $A \vee B \vee C$  вместо  $(A \vee B) \vee C$ .

### ЧАСТЬ 1

Ответами к заданиям 1-23 являются число, последовательность букв или цифр, которые следует записать в строке под заданием.

1. Сколько единиц в двоичной записи восьмеричного числа 17318?

Ответ запишите в строке под заданием

---

2. Логическая функция  $F$  задаётся выражением  $(x \vee y) \rightarrow (y \equiv z)$ . На рисунке приведён частично заполненный фрагмент таблицы истинности функции  $F$ , содержащий неповторяющиеся строки. Определите, какому столбцу таблицы истинности функции  $F$  соответствует каждая из переменных  $x, y, z$ .

?	?	?	<b>F</b>
<b>0</b>	<b>0</b>		<b>0</b>
<b>0</b>			<b>0</b>

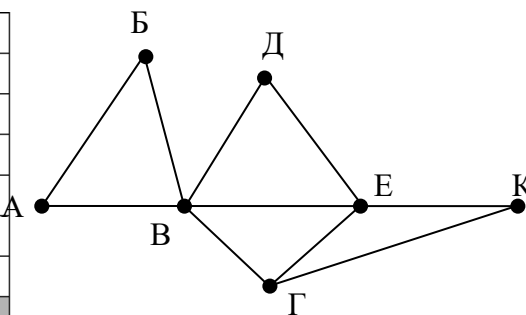
В ответе напишите буквы  $x, y, z$  в том порядке, в котором идут соответствующие им столбцы. Буквы в ответе пишите подряд, никаких разделителей между буквами ставить не нужно.

Ответ запишите в строке под заданием

3. На рисунке справа схема дорог Н-ского района изображена в виде графа, в таблице содержатся сведения о длинах этих дорог (в километрах). Так как таблицу и схему рисовали независимо друг от друга, то нумерация населённых пунктов в таблице никак не связана с буквенными обозначениями на графе. Определите, какова длина дороги из пункта В в пункт Е. В ответе запишите целое число – так, как оно указано в таблице.

Ответ запишите в строке под заданием

	П1	П2	П3	П4	П5	П6	П7
П1		45		10			
П2	45			40		55	
П3					15	60	
П4	10	40				20	35
П5			15			55	
П6		55	60	20	55		45
П7				35		45	



4. В фрагменте базы данных представлены сведения о родственных отношениях. На основании приведённых данных определите, сколько прямых потомков (т.е. детей и внуков) Павленко А.К. упомянуты в таблице 1.

Ответ запишите в строке под заданием

**Таблица 1**

ID	Фамилия_И.О.	Пол
2146	Кривич Л.П.	Ж
2155	Павленко А.К.	М
2431	Хитрук П.А.	М
2480	Кривич А.А.	М
2302	Павленко Е.А.	Ж
2500	Сокол Н.А.	Ж
3002	Павленко И.А.	М
2523	Павленко Т.Х.	Ж
2529	Хитрук А.П.	М
2570	Павленко П.И.	М
2586	Павленко Т.И.	Ж
2933	Симонян А.А.	Ж
2511	Сокол В.А.	Ж
3193	Биба С.А.	Ж

**Таблица 2**

ID_Родителя	ID_Ребенка
2146	2302
2146	3002
2155	2302
2155	3002
2302	2431
2302	2511
2302	3193
3002	2586
3002	2570
2523	2586
2523	2570
2529	2431
2529	2511

5. По каналу связи с помощью равномерного двоичного кода передаются сообщения, содержащие только 4 буквы: X, Y, Z, W; для кодировки букв используются кодовые слова длины 5. При этом для набора кодовых слов выполнено такое свойство: *любые два слова из набора отличаются не менее чем в трёх позициях*. Это свойство важно для

расшифровки сообщений при наличии помех. Для кодирования букв X, Y, Z используются 5-битовые кодовые слова:

X: 01111, Y: 00001, Z: 11000. Определите 5-битовое кодовое слово для буквы W, если известно, что оно начинается с 1 и заканчивается 0.

Ответ запишите в строке под заданием

---

6. На вход алгоритма подаётся натуральное число N. Алгоритм строит по нему новое число R следующим образом.

1) Строится двоичная запись числа N.

2) К этой записи дописываются справа ещё два разряда по следующему правилу:

а) складываются все цифры двоичной записи, и остаток от деления суммы на 2 дописывается в конец числа (справа). Например, запись 11100 преобразуется в запись 111001;

б) над этой записью производятся те же действия – справа дописывается остаток от деления суммы цифр на 2.

Полученная таким образом запись (в ней на два разряда больше, чем в записи исходного числа N) является двоичной записью искомого числа R. Укажите минимальное число N, после обработки которого с помощью этого алгоритма получается число, большее, чем 137. В ответе это число запишите в десятичной системе.

Ответ запишите в строке под заданием

---

7. Дан фрагмент электронной таблицы. Из ячейки D2 в одну из ячеек диапазона E1:E4 была скопирована формула. При копировании адреса ячеек в формуле автоматически изменились, и значение формулы стало равным 8. В какую ячейку была скопирована формула? В ответе укажите только одно число – номер строки, в которой расположена ячейка.

Ответ запишите в строке под заданием

	A	B	C	D	E
1	1	2	3	4	
2	2	3	4	=B\$3+\$C2	
3	3	4	5	6	
4	4	5	6	7	

---

8. Определите, что будет напечатано в результате работы следующего фрагмента программы.

Ответ запишите в строке под заданием



Бейсик	Алгоритмический язык	Python
<pre> DIM S, K AS INTEGER S = 0 K = 1 WHILE S &lt; 11 S = S + K K = K + 1 WEND PRINT S </pre>	<pre> алг нач цел k, s s := 0 k := 1 нц пока k &lt; 11 s := s + k k := k + 1 кц ВЫВОД s кО </pre>	<pre> s=0 k=1 while k &lt; 11: s=s+k k=k+1 print(s) </pre>
C++		Паскаль
<pre> #include &lt;iostream&gt; using namespace std; int main() { int k, s; s = 0; k = 1; while (k &lt; 11) { s = s + k; k = k + 1; } cout &lt;&lt; s; return 0; } </pre>		<pre> var k, s: integer; begin s:=0; k:=1; while k &lt; 11 do begin s:=s+k; k:=k+1; end; write(s); end. </pre>

9. Рисунок размером 512 на 256 пикселей занимает в памяти 64 Кбайт (без учёта сжатия). Найдите максимально возможное количество цветов в палитре изображения. Ответ запишите в строке под заданием

10. Все 4-буквенные слова, составленные из букв К, Л, Р, Т, записаны в алфавитном порядке и пронумерованы. Вот начало списка:

1. КККК
2. КККЛ
3. КККР
4. КККТ
- .....

Запишите слово, которое стоит на 67-м месте от начала списка.

Ответ запишите в строке под заданием

11. Ниже на пяти языках программирования записан рекурсивный алгоритм F.

Бейсик	Алгоритмический язык	Python
<pre>SUB F(n)   IF n &gt; 0 THEN     PRINT n     F(n - 3)     F(n \ 3)   END IF END SUB</pre>	<pre>алг F(цел n) нач   если n &gt; 0 то     вывод n     F(n - 3)     F(div(n, 3))   все кон</pre>	<pre>def F(n):   if n &gt; 0:     print(n)     F(n - 3)     F(n // 3)</pre>
C++		Паскаль
<pre>void F(int n){   if (n &gt; 0){     std::cout &lt;&lt;n;     F(n - 3);     F(n / 3);   } }</pre>		<pre>procedure F(n: integer); begin   if n &gt; 0 then     begin       write(n);       F(n - 3);       F(n div 3)     end end;</pre>

Запишите подряд без пробелов и разделителей все числа, которые будут напечатаны на экране при выполнении вызова F(9). Числа должны быть записаны в том же порядке, в котором они выводятся на экран.

Ответ запишите в строке под заданием

---

12. Два узла, находящиеся в одной сети, имеют IP-адреса 118.222.130.140 и 118.222.201.140. Укажите наибольшее возможное значение третьего слева байта маски сети. Ответ запишите в виде десятичного числа.

Ответ запишите в строке под заданием

---

13. При регистрации в компьютерной системе каждому пользователю выдаётся пароль, состоящий из 11 символов. Из соображений информационной безопасности каждый пароль должен содержать хотя бы 2 десятичных цифры, как прописные, так и строчные латинские буквы, а также не менее 2-х символов из 6-символьного набора: «&», «#», «\$», «\*», «!», «@». В базе данных для хранения сведений о каждом пользователе отведено одинаковое и минимально возможное целое число байт. При этом используют посимвольное кодирование паролей, все символы кодируют одинаковым и минимально возможным количеством бит. Кроме собственно пароля, для каждого пользователя в

системе хранятся дополнительные сведения, для чего выделено целое число байт; это число одно и то же для всех пользователей. Для хранения сведений о 30 пользователях потребовалось 900 байт. Сколько байт выделено для хранения дополнительных сведений об одном пользователе? В ответе запишите только целое число – количество байт.

Ответ запишите в строке под заданием

---

14. Редактор получает на вход строку цифр и преобразовывает её. Редактор может выполнять две команды, в обеих командах  $v$  и  $w$  обозначают цепочки цифр.

**А) заменить ( $v, w$ )**

Эта команда заменяет в строке первое слева вхождение цепочки  $v$  на цепочку  $w$ .

**Б) нашлось ( $v$ )**

Эта команда проверяет, встречается ли цепочка  $v$  в строке исполнителя Редактор. Если она встречается, то команда возвращает логическое значение «истина», в противном случае возвращает значение «ложь». Строка при этом не изменяется.

Дана программа для исполнителя Редактор:

**НАЧАЛО**

**ПОКА нашлось (222) ИЛИ нашлось (888)**

**ЕСЛИ нашлось (222)**

**ТО заменить (222, 8)**

**ИНАЧЕ заменить (888, 2)**

**КОНЕЦ ЕСЛИ**

**КОНЕЦ ПОКА**

**КОНЕЦ**

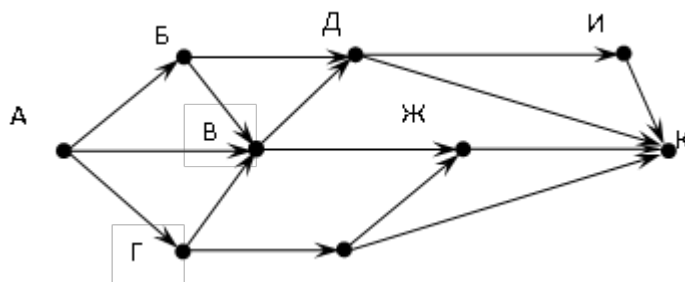
Какая строка получится в результате применения приведённой выше программы к строке, состоящей из 68 идущих подряд цифр 8? В ответе запишите полученную строку.

Ответ запишите в строке под заданием

---

15. На рисунке – схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, И, К. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город К?

Ответ запишите в строке под заданием



16. Значение арифметического выражения:  $9^9 - 3^9 + 9^{19} - 19$  записали в системе счисления с основанием 3. Сколько цифр «2» содержится в этой записи?

Ответ запишите в строке под заданием

17. В таблице приведены запросы и количество найденных по ним страниц некоторого сегмента сети Интернет:

Запрос	Количество страниц (тыс.)
Ростов & (Орёл & Курск   Белгород)	370
Ростов & Белгород	204
Ростов & Орёл & Курск & Белгород	68

Сколько страниц (в тысячах) будет найдено по запросу

***Ростов & Орёл & Курск***

Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

Ответ запишите в строке под заданием

18. Укажите наименьшее целое значение  $A$ , при котором выражение

$$(y + 2x < A) \vee (3y + 2x > 120) \vee (3y - x > 30)$$

истинно для любых целых положительных значений  $x$  и  $y$ .

Ответ запишите в строке под заданием

19. В программе используется одномерный целочисленный массив  $A$  с индексами от 0 до 9. Значения элементов равны 3, 0, 4, 6, 5, 1, 8, 2, 9, 7 соответственно, т.е.  $A[0] = 3$ ,  $A[1] = 0$  и т.д.

Определите значение переменной  $c$  после выполнения следующего фрагмента этой программы (записанного ниже на разных языках программирования).

Бейсик	Алгоритмический язык	Python
<pre> c = 0 FOR i = 1 TO 9     IF A(i-1) &gt; A(i) THEN         c = c + 1         t = A(i)         A(i) = A(i-1)         A(i-1) = t     END IF NEXT i                     </pre>	<pre> c := 0 нц для i от 1 до 9     если A[i-1] &gt; A[i]         то             c := c + 1             t := A[i]             A[i] := A[i-1]             A[i-1] := t         все кц                     </pre>	<pre> c = 0 for i in range(1,10):     if A[i-1] &gt; A[i]:         c = c + 1         t = A[i]         A[i] = A[i-1]         A[i-1] = t                     </pre>
<b>C++</b>		<b>Паскаль</b>
<code>c = 0;</code>		<code>c := 0;</code>

<pre> for (int i = 1; i &lt; 10; i++)     if (A[i-1] &gt; A[i]){         c++;         t = A[i];         A[i] = A[i-1];         A[i-1] = t;     } </pre>	<pre> for i := 1 to 9 do     if A[i-1] &gt; A[i] then     begin         c := c + 1;         t := A[i];         A[i] := A[i-1];         A[i-1] := t;     end; </pre>
---	---

Ответ запишите в строке под заданием

---

20. Ниже на пяти языках программирования записан алгоритм. Получив на вход число  $x$ , этот алгоритм печатает два числа:  $L$  и  $M$ . Укажите наименьшее число  $x$ , при вводе которого алгоритм печатает сначала 5, а потом 7.

Бейсик	Алгоритмический язык	Python
<pre> DIM X, L, M AS INTEGER INPUT X L = 0 M = 0 WHILE X &gt; 0     M = M + 1     IF X MOD 2 &lt;&gt; 0 THEN         L = L + 1     END IF X = X \ 2 WEND PRINT L PRINT M </pre>	<pre> алг нач     цел x, L, M     ввод x     L := 0     M := 0     нц пока x &gt; 0         M := M + 1         если mod(x,2) &lt;&gt; 0             то                 L := L + 1         все         x := div(x,2)     кц     вывод L, M кон </pre>	<pre> x = int(input()) L = 0 M = 0 while x &gt; 0:     M = M + 1     if x % 2 != 0:         L = L + 1     x = x // 2 print(L) print(M) </pre>
C++	Паскаль	
<pre> #include &lt;iostream&gt; using namespace std; int main(){     int x, L, M;     cin &gt;&gt; x;     L = 0;     M = 0;     while (x &gt; 0) {         M = M + 1;         if(x % 2 != 0) {             L = L + 1;         }     } } </pre>	<pre> var x, L, M: integer; begin     readln(x);     L := 0;     M := 0;     while x&gt;0 do     begin         M := M + 1;         if x mod 2 &lt;&gt; 0 then             L := L + 1;         x := x div 2;     end; end; </pre>	

<pre>         x = x / 2;     }     cout &lt;&lt; L &lt;&lt; endl &lt;&lt; M &lt;&lt; endl;     return 0; } </pre>	<pre>         writeln(L);         writeln(M);     end. </pre>
---	---

Ответ запишите в строке под заданием

21. Напишите в ответе число, которое будет напечатано в результате выполнения следующего алгоритма. Для Вашего удобства алгоритм представлен на пяти языках программирования.

Бейсик	Python
<pre> DIM A, B, T, M, R AS LONG A = -20: B = 20 M = A: R = F(A) FOR T = A TO B     IF F(T) &lt;= R THEN         M = T         R = F(T)     END IF NEXT T PRINT M+R FUNCTION F (x)     F = 2*(x*x-1)*(x*x-1)+27 END FUNCTION </pre>	<pre> def F(x):     return 2*(x*x-1)*(x*x-1)+27 a = -20; b=20 M=a; R=F(a) for t in range(a,b+1):     if (F(t) &lt;= R):         M=t; R=F(t) print (M+R) </pre>
Алгоритмический язык	Паскаль
<pre> алг нач цел a, b, t, M, R     a:=-20; b:=20     M:=a; R:=F(a)     нц для t от a до b         если F(t) &lt;= R то             M:=t; R:=F(t)         все     кц     вывод M+R кон алг цел F(цел x) нач     знач:=2*(x*x-1)*(x*x-1)+27 кон </pre>	<pre> var a, b, t, M, R :longint; function F(x: longint) : longint; begin     F:= 2*(x*x-1)*(x*x-1)+27; end; begin a:=-20; b:=20; M:=a; R:=F(a); for t:= a to b do begin     if (F(t) &lt;= R) then begin         M:=t;         R:=F(t)     end end; write(M+R) end. </pre>
C++	

```

#include <iostream>
using namespace std;
long F(long x) {
    return 2*(x*x-1)*(x*x-1)+27;
}
int main() {
    long a = -20, b = 20, M = a, R = F(a);
    for (int t = a; t <= b; ++t) {
        if (F(t) <= R) {
            M = t; R = F(t);
        }
    }
    cout << M + R;
    return 0;
}

```

Ответ запишите в строке под заданием

---

22. Исполнитель M17 преобразует число, записанное на экране. У исполнителя есть три команды, которым присвоены номера:

1. Прибавить 1
2. Прибавить 2
3. Умножить на 3

Первая из них увеличивает число на экране на 1, вторая увеличивает его на 2, третья умножает на 3.

Программа для исполнителя M17 – это последовательность команд.

Сколько существует таких программ, которые преобразуют исходное число 2 в число 12 и при этом траектория вычислений программы содержит числа 8 и 10? Траектория должна содержать оба указанных числа.

Траектория вычислений программы – это последовательность результатов выполнения всех команд программы. Например, для программы **132** при исходном числе 7 траектория будет состоять из чисел 8, 24, 26.

Ответ запишите в строке под заданием

---

23. Сколько различных решений имеет система логических уравнений

$$(x_1 \wedge y_1) \rightarrow (\neg x_2 \vee \neg y_2) = 1$$

$$(x_2 \wedge y_2) \rightarrow (\neg x_3 \vee \neg y_3) = 1$$

$$(x_3 \wedge y_3) \rightarrow (\neg x_4 \vee \neg y_4) = 1$$

$$(x_4 \wedge y_4) \rightarrow (\neg x_5 \vee \neg y_5) = 1$$

$$x_2 \wedge y_4 = 0$$

где  $x_1, \dots, x_5, y_1, \dots, y_5$  – логические переменные? В ответе не нужно перечислять все различные наборы значений переменных, при которых выполнено данное равенство. В качестве ответа нужно указать количество таких наборов.

Ответ запишите в строке под заданием

## ЧАСТЬ 2

24. Требовалось написать программу, в которой нужно было проверить, лежит ли число  $x$  на числовой оси между числами  $a$  и  $b$  ("между" понимается в строгом смысле, т.е. случай  $x=a$  или  $x=b$  недопустим). Числа  $x, a, b$  являются натуральными, и известно, что  $a$  отлично от  $b$  (но неизвестно:  $a>b$  или  $b>a$ ). Входная информация вводится с клавиатуры, а на выходе должно быть сообщение вида " $x$  между  $a$  и  $b$ " (если это действительно так), в противном случае никакой выходной информации не выдается.

Программист торопился и написал программу некорректно.

Паскаль	Бейсик
<pre>VAR a,b,x: integer; p: integer; BEGIN readln(a,b,x); if (a&gt;x) AND (x&gt;b) then writeln('x между a,b'); END.</pre>	<pre>CLS INPUT a, b, x IF (a&gt;x) AND (x&gt;b) THEN PRINT "x между a, b" END</pre>

Последовательно выполните три задания.

- 1) Приведите пример таких чисел  $a, b, x$ , при которых программа работает неправильно.
- 2) Укажите, как нужно доработать программу, чтобы не было случаев ее неправильной работы. (Это можно сделать несколькими способами, поэтому можно указать любой способ доработки исходной программы).
- 3) Укажите, как можно доработать программу, соблюдая дополнительное условие: доработанная программа не должна использовать логических операций AND или OR.

Ответы впишите в поле под заданиями

1.
2.
3.



25. Опишите на русском языке или на одном из языков программирования алгоритм поиска второго по величине (т.е. следующего по величине за максимальным) элемента в числовом массиве из 30 различных элементов.

Ответ впишите в поле под заданием

26. Два игрока играют в следующую игру. Перед ними лежат две кучки камней, в первой из которых 5, а во второй – 3 камня. У каждого игрока неограниченно много камней. Игроки ходят по очереди. Ход состоит в том, что игрок или удваивает число камней в какой-то куче, или добавляет 4 камня в какую-то кучу. Выигрывает игрок, после хода которого в одной из куч становится не менее 22 камней. Кто выигрывает при безошибочной игре обоих игроков – игрок, делающий первый ход, или игрок, делающий второй ход? Как должен ходить выигрывающий игрок? Ответ обоснуйте.

Ответ впишите в поле под заданием

27. Вступительные испытания в некоторый вуз состоят из трех экзаменов: математика (максимальный балл – 9), информатика (максимальный балл – 9), литература (максимальный балл – 5). На вход программе подаются сведения о сдаче этих экзаменов абитуриентами. В первой строке вводится количество абитуриентов  $N$ , во второй – количество мест  $K$  ( $K < N$ ) на которые эти абитуриенты претендуют. Каждая из следующих  $N$  строк имеет следующий формат: <Фамилия> <оценка1> <оценка2> <оценка3>, где <Фамилия> – строка, состоящая не более, чем из 20 символов, оценки – числа от 0 до максимальной оценки по предмету соответственно. (Ноль ставится в случае, если экзамен не сдавался, например, после полученной на предыдущем экзамене двойки. Все баллы, большие 2, считаются удовлетворительными). Пример входных строк:

Иванов 8 9 3

Петров 2 0 0

Требуется написать программу на языке Паскаль или Бейсик, которая определяла бы по имеющимся данным количество абитуриентов, набравших полупроходной балл в данный вуз или сообщала, что такой балл отсутствует. (Полупроходным называется такой балл, что лишь часть абитуриентов, набравших такой балл и не получивших ни одной неудовлетворительной оценки, попадает в  $K$  лучших, которые должны быть

зачислены на 1 курс) Считается, что абитуриенты, получившие только удовлетворительные оценки, обязательно присутствуют.

Ответ впишите в поле под заданием

## ТАБЛИЦА ОТВЕТОВ (КЛЮЧИ)

### ЧАСТЬ 1

За правильный ответ на задания 1-23 ставится 3 балла; за неверный ответ или его отсутствие – 0 баллов.

№ задания	Ответ
1	7
2	zxy
3	20
4	7
5	10110
6	35
7	1
8	55
9	16
10	ЛККР
11	9631231
12	128
13	20
14	28
15	13
16	16
17	234
18	1188
19	5
20	79
21	28
22	60
23	256

Максимальный балл за Часть 1 – 69 баллов

### ЧАСТЬ 2

№ задания	Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)	Баллы
-----------	--	-------

№ задания	Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)	Баллы
24	<p>Элементы ответа:</p> <p>1) Пример: a=1 x=2 b=3</p> <p>2) Возможная доработка:  if a&lt;b then begin p:=a; a:=b; b:=p end;  if (a&gt;x) AND (x&gt;b) then  writeln(' x между a,b');  (могут быть и другие правильные способы доработки).</p> <p>3) Возможная доработка без использования логических операций AND, OR:  p:=(x-a)*(x-b); if p&lt;0 then  writeln(' x между a,b');  (могут быть и другие способы доработки с соблюдением дополнительного условия).</p> <p>При оценке других вариантов доработки программы нужно проверять, что поставленная цель достигается.</p>	
<b>Указания по оцениванию</b>		
	Правильно выполнены п.1)+п.3) задания (т.к. выполнение п.3 "покрывает" и пункт 2), или правильно выполнены все 3 пункта задания, при этом в работе (во фрагментах программ) допускается не более одной пунктуационной ошибки	6
	Правильно выполнены 2 пункта задания: 1)+2) или 2)+3), (причем способы доработки в п.2 и п.3 различны). При этом в сданной работе допускается не более двух синтаксических ошибок (пропущен или неверно указан знак пунктуации, неверно написано зарезервированное слово языка программирования)	4
	Правильно выполнен только один пункт задания, при этом, если это был п.2) или п.3), то в нем допускается не более двух синтаксических ошибок (пропущен или неверно указан знак пунктуации, неверно написано зарезервированное слово языка программирования)	2
	Все пункты задания выполнены неверно	0
<i>Максимальный балл</i>		6
25	Введем числовые переменные <i>Max1</i> и <i>Max2</i> , в которых будем хранить соответственно максимальный и следующий за максимальным элемент в уже просмотренной части массива. Затем в цикле до конца массива сравниваем очередной элемент массива с двумя максимальными, и если он больше одного из них или обоих, то меняем два отобранных элемента. По окончании цикла	

№ задания	Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)	Баллы				
	переменная <i>Max2</i> содержит второй по величине элемент массива.					
<b>Указания по оцениванию</b>						
	<p>Предложен правильный алгоритм, выдающий верное значение (в том числе и алгоритм, требующий двукратного прохода по массиву).</p> <p>Возможно использование числа 30 вместо константы. Возможно наличие отдельных синтаксических ошибок (пропущенные «;», неверная запись оператора присваивания и т.п.), не искажающих замысла автора программы.</p> <p>В качестве примера правильного и эффективного алгоритма приведен фрагмент программы:</p> <table border="1" data-bbox="284 797 1310 1715"> <thead> <tr> <th data-bbox="284 797 799 846">На языке Паскаль</th> <th data-bbox="807 797 1310 846">На языке Бейсик</th> </tr> </thead> <tbody> <tr> <td data-bbox="284 853 799 1715"> <pre>const N=30; var a:array[1..N] of real;     Max1, Max2, i: real; begin   Max1:=a[1];   Max2:=a[1];   if a[2]&gt;Max1 then Max1:=a[2]   else Max2:=a[2];   for i:=3 to N do   begin     if a[i]&gt;Max1 then     begin Max2:=Max1;       Max1:=a[i];     end     else if a[i]&gt;Max2 then       Max2:=a[i];     end;   writeln(Max2); end.</pre> </td> <td data-bbox="807 853 1310 1715"> <pre>N=30 DIM i, Max1, Max2, a(N) AS REAL Max1=a(1) Max2=a(1) IF a(2)&gt;Max1 THEN Max1=a(2) ELSE Max2=a(2) FOR i = 3 TO N IF a(i)&gt;Max1 THEN   Max2=Max1   Max1=a(i) ELSE   IF a(i)&gt;Max2 THEN     Max2=a(i)   ENDIF ENDIF NEXT i PRINT Max2 END</pre> </td> </tr> </tbody> </table>	На языке Паскаль	На языке Бейсик	<pre>const N=30; var a:array[1..N] of real;     Max1, Max2, i: real; begin   Max1:=a[1];   Max2:=a[1];   if a[2]&gt;Max1 then Max1:=a[2]   else Max2:=a[2];   for i:=3 to N do   begin     if a[i]&gt;Max1 then     begin Max2:=Max1;       Max1:=a[i];     end     else if a[i]&gt;Max2 then       Max2:=a[i];     end;   writeln(Max2); end.</pre>	<pre>N=30 DIM i, Max1, Max2, a(N) AS REAL Max1=a(1) Max2=a(1) IF a(2)&gt;Max1 THEN Max1=a(2) ELSE Max2=a(2) FOR i = 3 TO N IF a(i)&gt;Max1 THEN   Max2=Max1   Max1=a(i) ELSE   IF a(i)&gt;Max2 THEN     Max2=a(i)   ENDIF ENDIF NEXT i PRINT Max2 END</pre>	6
На языке Паскаль	На языке Бейсик					
<pre>const N=30; var a:array[1..N] of real;     Max1, Max2, i: real; begin   Max1:=a[1];   Max2:=a[1];   if a[2]&gt;Max1 then Max1:=a[2]   else Max2:=a[2];   for i:=3 to N do   begin     if a[i]&gt;Max1 then     begin Max2:=Max1;       Max1:=a[i];     end     else if a[i]&gt;Max2 then       Max2:=a[i];     end;   writeln(Max2); end.</pre>	<pre>N=30 DIM i, Max1, Max2, a(N) AS REAL Max1=a(1) Max2=a(1) IF a(2)&gt;Max1 THEN Max1=a(2) ELSE Max2=a(2) FOR i = 3 TO N IF a(i)&gt;Max1 THEN   Max2=Max1   Max1=a(i) ELSE   IF a(i)&gt;Max2 THEN     Max2=a(i)   ENDIF ENDIF NEXT i PRINT Max2 END</pre>					
	<p>Имеется не более двух ошибок из числа следующих:</p> <ol style="list-style-type: none"> <li>1) Не задано или неверно задано первое значение <i>Max1</i></li> <li>2) Неверно вычисляется первое значение переменной <i>Max2</i></li> <li>3) Не указано условие завершения цикла</li> <li>4) Программа не выводит результат</li> <li>5) Индексная переменная в цикле не увеличивается (при использовании циклов <code>while</code> или <code>repeat-until</code>)</li> </ol> <p>В программе на Паскале неверно расставлены операторные скобки</p>	3				

<b>№ задания</b>	<b>Содержание верного ответа и указания по оцениванию</b> (допускаются иные формулировки ответа, не искажающие его смысла)	<b>Баллы</b>																																				
	Ошибок, перечисленных выше, больше двух или алгоритм сформулирован неверно (в частности, не хранится следующий за максимальным элемент)	0																																				
<i>Максимальный балл</i>		6																																				
26	<p>Выигрывает первый игрок. Своим первым ходом он должен удвоить количество камней во второй куче. Для доказательства рассмотрим неполное дерево игры после этого хода первого игрока.</p> <table border="1" data-bbox="284 707 1315 1585"> <thead> <tr> <th data-bbox="284 707 459 891">Позиция после первого хода</th> <th data-bbox="459 707 628 891">1-й ход второго игрока</th> <th data-bbox="628 707 935 891">Выигрывающий ход первого игрока</th> <th data-bbox="935 707 1315 891">Пояснение</th> </tr> </thead> <tbody> <tr> <td data-bbox="284 891 459 936"></td> <td data-bbox="459 891 628 936"></td> <td data-bbox="628 891 935 936"></td> <td data-bbox="935 891 1315 936"></td> </tr> <tr> <td data-bbox="284 936 459 1211"></td> <td data-bbox="459 936 628 1211">10,6</td> <td data-bbox="628 936 935 1211">10,10</td> <td data-bbox="935 936 1315 1211">Первый игрок выигрывает после любого ответа второго игрока, удвоив число камней в самой большой куче</td> </tr> <tr> <td data-bbox="284 1211 459 1256"></td> <td data-bbox="459 1211 628 1256"></td> <td data-bbox="628 1211 935 1256"></td> <td data-bbox="935 1211 1315 1256"></td> </tr> <tr> <td data-bbox="284 1256 459 1352">5,6</td> <td data-bbox="459 1256 628 1352">9,6</td> <td data-bbox="628 1256 935 1352">9,10</td> <td data-bbox="935 1256 1315 1352">"-"</td> </tr> <tr> <td data-bbox="284 1352 459 1397"></td> <td data-bbox="459 1352 628 1397"></td> <td data-bbox="628 1352 935 1397"></td> <td data-bbox="935 1352 1315 1397"></td> </tr> <tr> <td data-bbox="284 1397 459 1449"></td> <td data-bbox="459 1397 628 1449">5,10</td> <td data-bbox="628 1397 935 1449">9,10 или 10,10</td> <td data-bbox="935 1397 1315 1449">"-"</td> </tr> <tr> <td data-bbox="284 1449 459 1494"></td> <td data-bbox="459 1449 628 1494"></td> <td data-bbox="628 1449 935 1494"></td> <td data-bbox="935 1449 1315 1494"></td> </tr> <tr> <td data-bbox="284 1494 459 1585"></td> <td data-bbox="459 1494 628 1585">5,12</td> <td data-bbox="628 1494 935 1585">5,24</td> <td data-bbox="935 1494 1315 1585">Выигрыш первого игрока</td> </tr> </tbody> </table> <p data-bbox="284 1637 1315 1765">Из таблицы видно, что при первом ходе (5,3)-&gt;(5,6) первый игрок выигрывает не позже, чем на третьем ходу при любом ответе второго игрока.</p>	Позиция после первого хода	1-й ход второго игрока	Выигрывающий ход первого игрока	Пояснение						10,6	10,10	Первый игрок выигрывает после любого ответа второго игрока, удвоив число камней в самой большой куче					5,6	9,6	9,10	"-"						5,10	9,10 или 10,10	"-"						5,12	5,24	Выигрыш первого игрока	
Позиция после первого хода	1-й ход второго игрока	Выигрывающий ход первого игрока	Пояснение																																			
	10,6	10,10	Первый игрок выигрывает после любого ответа второго игрока, удвоив число камней в самой большой куче																																			
5,6	9,6	9,10	"-"																																			
	5,10	9,10 или 10,10	"-"																																			
	5,12	5,24	Выигрыш первого игрока																																			
<b>Указания по оцениванию</b>																																						
	Правильное указание игрока и его ходов со строгим доказательством правильности с помощью или без помощи дерева игры	9																																				
	Правильное указание выигрывающего игрока, стратегии игры, приводящей к победе, но при отсутствии доказательства ее	6																																				

№ задания	Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)	Баллы
	правильности.	
	<p>При наличии в представленном решении одного из пунктов:</p> <p>1. Правильно указаны все варианты хода первого игрока и возможные ответы второго игрока (в том числе, и все выигрышные), но неверно определены дальнейшие действия и неправильно указан победитель.</p> <p>2. Правильно указан выигрывающий игрок, но отсутствует описание выигрышной стратегии и рассмотрены не все варианты хода первого игрока (а только один или несколько) и частные случаи ответов второго игрока.</p>	3
	Задание не выполнено или в представленном решении полностью отсутствует описание элементов выигрышной стратегии, и отсутствует анализ вариантов первого и второго ходов играющих.	0
	<i>Максимальный балл</i>	9
27	<p>Программа верно читает входные данные, не запоминая их все, а сразу подсчитывая в массиве, хранящем 24 целых числа, количество абитуриентов, набравших тот или иной балл (от 0 до 23). Если при этом абитуриент получил хотя бы одну двойку, то удобно считать, что его общий балл равен 0. Затем вычисляется сумма элементов этого массива, начиная с 23-го, до тех пор пока она не превосходит K. Индекс первого элемента массива, который не войдет в эту сумму и будет искомым полупроходным баллом. Если проходной балл набрали ровно K абитуриентов, то программа сообщает, что полупроходной балл отсутствует. Баллы начисляются только за программу, которая решает задачу хотя бы для частного случая (например, проходной балл набрали строго меньше K абитуриентов).</p>	
<b>Указания по оцениванию</b>		
	Программа работает верно, т.е. корректно выделяет из входных данных оценки абитуриентов, верно учитывает результаты абитуриентов, получивших двойки, не содержит вложенных циклов (от 1 до N и от 0 до 23). Допускается наличие в тексте программы одной пунктуационной ошибки.	10
	<p>Пример правильной и эффективной программы на языке Паскаль:</p> <pre>var m:array[0..23] of integer;     c:char;     i, K, N, S, m1, m2, m3:integer; begin</pre>	

№ задания	Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)	Баллы
	<pre> readln(N); readln(K); for i:=0 to 23 do m[i]:=0; for i:=1 to N do begin repeat read(c) until c=' '; {считана фамилия абитуриента} readln(m1, m2, m3); if (m1&lt;3)or(m2&lt;3)or(m3&lt;3) then s:=0 else s:=m1+m2+m3; m[s]:=m[s]+1 {учитываем абитуриента в элементе массива, соответствующем его баллам} end; s:=m[23]; i:=23; while s+m[i-1]&lt;=K and (i&gt;9) {9 - минимально возможный балл} do begin i:=i-1; s:=s+m[i] end; if (s&lt;K)and(i&gt;9) then writeln('полупроходной балл набрали', m[i-1], ' человек') else writeln('полупроходной балл отсутствует'); readln end. </pre>	
	<p>Пример правильной программы на языке Бейсик:</p> <pre> DIM i, j, k, n, m1, m2, m3, s, m(23) AS INTEGER DIM ss AS STRING FOR i = 1 TO 23 m(i) = 0 NEXT i INPUT n INPUT k FOR j = 1 TO n LINE INPUT ss i = 1 c\$ = MID\$(ss, i, 1) </pre>	



№ задания	Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)	Баллы
	<pre> WHILE NOT (c\$ = " ") i = i + 1 c\$ = MID\$(ss, i, 1) WEND ss = MID\$(ss, i + 1, 5) m1 = ASC(MID\$(ss, 1, 1)) - ASC("0") m2 = ASC(MID\$(ss, 3, 1)) - ASC("0") m3 = ASC(MID\$(ss, 5, 1)) - ASC("0") IF (m1 &lt; 3) OR (m2 &lt; 3) OR (m3 &lt; 3) THEN s = 1 ELSE s = m1 + m2 + m3 END IF m(s) = m(s) + 1 NEXT j s = m(23): i = 23 WHILE (s + m(i - 1) &lt;= k) AND (i &gt; 9) i = i - 1 s = s + m(i) WEND IF (s &lt; k) AND (i &gt; 9) THEN PRINT "Полупроходной балл набрали"; m(i - 1);" человек" ELSE PRINT "Полупроходной балл отсутствует" END IF END </pre>	
	<p>Программа работает верно, но содержит вложенные циклы (от 0 до 23 и от 1 до N) или несколько операторов IF (по количеству возможных баллов у абитуриента) или оператор CASE, обрабатывающий различные варианты количества баллов абитуриента. Возможно, сохраняет все входные данные в массиве абитуриентов. Допускается наличие от одной до трех различных синтаксических ошибок: пропущен или неверно указан знак пунктуации, неверно написано зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, недопустимая для соответствующего типа данных.</p>	8
	<p>Программа не учитывает случай, когда ровно K абитуриентов набрали проходной балл или что количество абитуриентов, получивших удовлетворительные оценки, может оказаться меньше K. Возможно, в реализации алгоритма содержатся 1–2 ошибки</p>	6

<b>№ задания</b>	<b>Содержание верного ответа и указания по оцениванию</b> (допускаются иные формулировки ответа, не искажающие его смысла)	<b>Баллы</b>
	(используется знак “<” вместо “<=”, “or” вместо “and”, выражение на 1 отличается от верного и т.п.). Допускается наличие от одной до пяти различных синтаксических ошибок.	
	Программа неверно работает при некоторых входных данных, возможно, содержит ошибку при учете баллов абитуриентов, получивших неудовлетворительные оценки или в выделении оценок из строки входных данных или в логике определения полупроходного балла. Допускается наличие от одной до семи различных синтаксических ошибок.	3
	Задание выполнено неверно	0
<i>Максимальный балл</i>		10

Максимальный балл за Часть 2 – 31 баллов