



УТВЕРЖДАЮ

Директор АНОО «Физтех-лицей»
им. П.Л. Капицы

Машкова М.Г.

« 01 » / 04 2024г.

ЭКЗАМЕНАЦИОННЫЕ ТРЕБОВАНИЯ

ИНФОРМАТИКА

2023-2024 учебный год

ИНФОРМАЦИОННО-ТЕХНОЛОГИЧЕСКИЙ ПРОФИЛЬ

10 класс

I. Форма проведения экзамена

Экзамен проводится в письменном формате. Будут предлагаться теоретические вопросы (в том числе «Приведите определение», «Докажите» и т.д.) и задачи по вопросам из приведённого ниже списка. Общее количество заданий – высокое (более 20).

II. Список вопросов

2.1. Системы счисления и компьютерная арифметика

1. Системы счисления: основные определения (базис, основание, алфавит и т.д.)
2. Свёрнутая и развёрнутая форма записи числа. Представление целых и рациональных чисел. Свойства числа, зависящие и не зависящие от системы счисления с обоснованием.
3. Алгоритм перечисления натуральных чисел. Примеры задач на его использование
4. Арифметические операции в позиционных системах счисления. Алгоритмы заполнения таблиц сложения и умножения.
5. Алгоритм перевода с использованием арифметики целевой системы счисления (для рациональных чисел) с обоснованием и реализацией на C++.
6. Алгоритм перевода периодических дробей с использованием арифметики целевой системы счисления с обоснованием и реализацией на C++.
7. Алгоритмы перевода целых чисел с использованием арифметики исходной системы счисления: делением и выделением максимальной степени с обоснованием. Их сравнительный анализ. Реализация на C++.
8. Алгоритм перевода рациональных чисел с использованием арифметики исходной системы счисления с обоснованием и реализацией на C++.
9. Смешанные системы счисления. Быстрый алгоритм перевода для систем счисления, основания которых связаны соотношением $r^m=q$ с обоснованием.
10. Нетрадиционные системы счисления. Расчёт количества цифр в разряде по базису. Избыточность представления.
11. Факториальная система счисления. Алгоритмы перевода чисел из факториальной системы в десятичную и обратно.
12. Фибоначчиева система счисления. Алгоритмы перевода чисел из фибоначчиевой системы в десятичную и обратно.

13. Уравновешенная троичная система счисления. Алгоритмы перевода из уравновешенной троичной системы счисления в десятичную и обратно.
14. Представление целых чисел в компьютере. Знаковый и беззнаковый формат.
15. Прямой и дополнительный код. Алгоритмы получения дополнительного кода по прямому и обратно.
16. Особенности целочисленной компьютерной арифметики в ограниченном количестве разрядов. Устройство целых типов.
17. Представление вещественных чисел в форматах с фиксированной и плавающей точкой. Достоинства и недостатки каждого представления.
18. Структура вещественного типа. Особенности записи мантиссы и показателя. Вещественные типы float и double. Точность вычислений.
19. Арифметические операции в формате с плавающей точкой. Ошибки вещественной арифметики ($A+B=A$, $A*B=0$ и т.д.)
20. Сравнение вещественных чисел.

2.2 Алгебра логики

1. Булевы функции (определение, примеры, расчёт количества).
2. Понятие ДНФ и КНФ. Теоремы о Совершенной ДНФ и Совершенной КНФ с доказательством.
3. Алгоритмы построения Совершенной ДНФ и Совершенной КНФ с обоснованием.
4. Минимальная ДНФ. Алгоритм минимизирующих карт с обоснованием.
5. Полные системы булевых функций. Стрелка Пирса, Штрих Шеффера – доказательство их полноты.
6. Логические элементы. Схемы из функциональных элементов. Устройство сумматора.

2.3. Теория алгоритмов

1. Понятие и свойства алгоритма (детерминированность, понятность, массовость, сложность).
2. Машина Тьюринга как способ записи алгоритма. Принципы работы. Правила оформления программ. Примеры программ на машине Тьюринга.
3. Тезис Тьюринга. Понятие алгоритмически неразрешимой задачи. Примеры.
4. Нормальные алгоритмы Маркова. Принципы работы. Правила записи программ. Примеры программ.
5. Понятие функции сложности алгоритма. Сложность в худшем и лучшем случаях. Оценки функции сложности $O(f)$, $\Omega(f)$, $\Theta(f)$. Примеры.
6. Оценка сложности рекурсивных алгоритмов. Метод оценки через дерево рекурсии.
7. Master theorem об оценке сложности, заданной рекуррентным соотношением с доказательством. Примеры применения.
8. Понятие конечного автомата. Принцип его работы. Детерминированные и недетерминированные конечные автоматы. Алгоритм детерминизации.

2.4. Язык программирования C/C++

1. Синтаксические диаграммы. Правила их построения. Примеры синтаксических диаграмм для описания понятий языка программирования (идентификатор, целое число, вещественное число в формате фиксированной точкой, вещественное число в формате с плавающей точкой, оператор, программа, описание переменных).
2. Структура программы на языке C++.

3. Понятие переменной. Правила именования. Описание.
4. Понятие типа переменной. Базовые типы в языке C++.
5. Логический тип bool. Преобразование чисел в логику и обратно. Расширенное понимание истинности.
6. Операции в C++. Вычисление операций. Приоритет операций. Тип результата. Преобразование типов при вычислениях. Операция явного приведения к типу.
7. Категории значений lvalue и rvalue. Операции, требующие lvalue, в качестве операнда (операндов). Операции, возвращающие lvalue в качестве значения.
8. Условный оператор (грамматика, как работает, особенности). Примеры задач с его использованием. Вложенные условные операторы. Отличия условного оператора от тернарной условной операции.
9. Операторы цикла (while, do while, for) – грамматика, как работает, особенности выполнения, примеры задач на использование каждого вида оператора.
10. Операторы break и continue. Отличия в работе данных операторов в зависимости от вида оператора цикла.
11. Правила оформления программ: принципы форматирования, именования переменных и т.д..
12. Кодовые таблицы. Таблица ASCII. Работа с символами.
13. Битовые операции. Реализуемые алгоритмы. Маски. Шестнадцатеричные литералы.
14. Указатели в C. Описание указателей. Операции разыменования, взятия адреса и копирования. Приоритет операций * и &.
15. Адресная арифметика. Сравнение указателей.
16. Массивы в C. Индексирование. Указатель текущей позиции.
17. Функции в C. Структура функции. Формальные и фактические параметры. Описание и вызов функции. Оператор return. Особенности возврата результата. Функции типа void.
18. Квалификаторы const и volatile. Отличие указателя на константу от константного указателя.
19. Принцип локализации имён.
20. Виды связывания переменных (без связывания, внутреннее, внешнее). Использование спецификатора static для изменения вида связывания.
21. Классификация переменных по классам памяти: Automatic, Static, Thread, Dynamic. Использование спецификатора static для изменения класса памяти.
22. Механизмы передачи параметров в функции: по значению, через указатель, по ссылке (C++ only).
23. Передача массивов в функции в качестве параметров. Отличия обычных массивов от std::vector.
24. Формальные параметры со значениями по умолчанию.
25. Прототипы функций. Перегрузка функций. Inline функции.
26. Динамические массивы. Понятия Size и Capacity. Реаллокация. Итераторы. Контейнеры vector и string в C++. Отличия string от C- строк.
27. Управление динамической памятью. Операторы new и delete.
28. Механизм обработки исключительных ситуаций в C++. try-блок. Вложенные try-блоки. Повторная генерация исключений.
29. Поточный ввод вывод в C++. Работа с файлами в C++.

2.5. Объектно-ориентированное программирование (ООП) – level 1

1. Общее описание парадигмы ООП. Заповеди ООП.
2. Классы и объекты. Описание. Примеры.
3. Виды членов класса. Описание методов внутри и вне класса. Примеры.
4. Доступ к членам класса. Спецификаторы доступа.
5. Статические члены класса.

6. Конструкторы и деструкторы. Виды конструкторов. Предназначение конструкторов и деструкторов.
7. Указатель `this`.
8. Конструктор по умолчанию. Неявный конструктор по умолчанию. Параметризованный конструктор по умолчанию.
9. Конструктор копирования. Ситуации срабатывания. Неявный конструктор копирования.
10. Функции `друзья`.
11. Перегрузка операций. Перегрузка операций членами классов. Перегрузка операций функциями-друзьями.

2.6. Рекурсия. Рекурсивные алгоритмы.

1. Понятие рекурсии. Рекурсивные определения. Прямая и косвенная рекурсия. Примеры.
2. Рекурсивные алгоритмы. Принципы написания рекурсивных функций.
3. Процесс рекурсивного вычисления. Примеры задач с тривиальным и нетривиальным процессом. Ханойские башни.
4. Волновой алгоритм Ли.
5. Рекурсивные сортировки: слияние, Хоара.
6. Рекурсивная схема перебора с возвратом.
7. Нотационные формы Бэкуса-Наура. Рекурсивный разбор строк по грамматикам.

2.7. Структуры данных

1. Понятие структуры данных. Виды структур данных. Реализация структур данных в виде массивов, и в виде списков (на указателях).
2. Списочные структуры данных. Однонаправленные и двунаправленные списки. Реализация в языке программирования на указателях. Операции обхода, вставки и удаления элементов (рекурсивно и нерекурсивно). Сложность операций.
3. Линейные структуры данных: стек, очередь, дек. Операции с ними. Реализации при помощи массива и на указателях. Сложность операций.
4. Алгоритм Дейкстры «Сортировочная станция». Применения на задачах различных классов.
5. Амортизационный анализ. Оценка сложности операции в цепочке методом усреднения. Банковский метод. Доказательство амортизированной сложности операции `push_back` у `std::vector`. Другие примеры.
6. Определение дерева. Основные понятия: корень, лист, обрыв, узел, родитель, ребенок, предок, потомок, высота. Бинарные деревья. Реализация в языке программирования на указателях.
7. Обходы бинарного дерева (рекурсивные и нерекурсивные): `preorder`, `inorder`, `postorder`, `levelorder`.
8. Деревья поиска. Свойства. Операции вставки и удаления элементов. Сложность операций.
9. AVL деревья. Свойства. Операции вставки и удаления элементов. Сложность операций. Балансировка AVL-деревьев.
10. Операции `split` и `merge` на AVL-деревьях. Сложность операций.
11. Совершенное дерево и дерево Фибоначчи. Свойства.
12. Красно-чёрные деревья. Свойства. Операции вставки и удаления элементов. Сложность операций. Балансировка Красно-чёрных деревьев.
13. Операции `split` и `merge` на Красно-чёрных деревьях. Сложность операций.

14. Структура данных двоичная куча. Реализация на массиве. Операции с кучей, их сложность.
15. Heap sort.
16. Декартово дерево. Построение дерева из списка значений. Операции разделения и слияния. Вставка, удаление элементов. Дуча (случайные приоритеты) и ее особенности, применимость. Оценки сложностей.
17. Декартово дерево по неявному ключу. Слияние и разделение. Задача о циклическом сдвиге за $O(\log n)$.
18. Splay-деревья. Реализации операций splay, find, merge, split, add, remove. Амортизированная оценка сложности операций.
19. Синтаксическое дерево. Построение синтаксического дерева. Использование синтаксического дерева для вычисления выражений.

2.8. Численные методы

1. Понятие численного метода для решения задач.
2. Задача вычисления определённого интеграла. Методы прямоугольников, трапеций и Симпсона. Правило Рунге.
3. Задача поиска корней уравнений на отрезке. Методы бинарного поиска, хорд и касательных (Ньютона).
4. Рандомизированные алгоритмы. Метод Монте-Карло для вычисления определённого интеграла и иррациональных констант.

2.9. Фракталы

1. Понятие фрактала. Виды фракталов. Применение фракталов.
2. Геометрические фракталы. Примеры геометрических фракталов (кривая Коха, снежинка Коха, треугольник Серпинского, ковёр Серпинского).
3. Алгебраические фракталы. Примеры алгебраических фракталов (множество Мандельброта, множество Жюлиа).
4. Стохастические фракталы. Примеры Стохастических фракталов (дерево, «Плазма», горный ландшафт).