

ЗАДАЧИ К ЛАБОРАТОРНЫМ РАБОТАМ ПО ООП (ч. 1, 52 ч.)

Алгоритмы простой структуры на языке C++ (12 ч.)

1. Простые вычисления по формуле. Составление *программы* вычисления и вывода на экран заданной формулы в виде функции $f(x)$, где $x \in [a; b]$ (задача 1.1).
2. Нахождение суммарных значений в цикле. Составление *программы* для вычисления в цикле суммарного результата и вывода его на экран (задача 1.3.3).
3. Нахождение экстремальных элементов в одномерном массиве. Составление *программы* нахождения экстремального элемента и/или его порядкового номера в одномерном массиве (или массивах) из n элементов и вывода результата поиска (задача 1.4.3).
4. Вычисление корня функционального уравнения. Разработка *программы* для решения уравнения $f(x) = 0$ на отрезке $[a; b]$ с заданной точностью методом деления отрезка пополам и вывода решения на экран (задача 1.7).
5. Табулирование функции с выводом результата по логическому условию. Составление *программы*, выводящей на экран таблицу значений функции, которые удовлетворяют заданному сложному условию (задача 1.9.3).

Алгоритмы более сложной структуры на языке C++ (12 ч.)

6. Обработка матриц с небольшим фиксированным числом строк и столбцов. Составление *программы* для решения задачи с использованием одномерных и/или двумерных массивов и вывода результатов решения на экран (задача 2.5.1).
7. Более сложное сочетание данных разной структуры. Составление *программы*, оперирующей с данными различной структуры (задача 2.5.4).
8. Табулирование функции двух аргументов. Разработка *программы* вычисления и вывода на экран таблицы значений функции двух аргументов с использованием матрицы для хранения значений функции (задача 3.2).

Функции в программах на языке C++ (12 ч.)

Замечание. Здесь и далее консольный ввод-вывод в программах следует выполнять на русском языке, используя преобразование кодировок. При этом желательно применять стандартный потоковый ввод-вывод в стиле языка C++.

9. Вычисление выражения с использованием одной функции. Разработка *программы*, использующей функцию для организации вычислений по заданной формуле. Эта функция для вычислений по заданной формуле должна вызываться в программе более одного раза и должна использовать во время своего выполнения только свои локальные переменные (задача 4.1).
10. Вычисление корня функционального уравнения при разных значениях точности. Разработка *программы*, содержащей функцию для вычисления корня заданного уравнения для заданного значения точности (задача 4.2.2).
11. Разработка программы с использованием созданной по условию функции. Составление *программы* с разработкой и использованием в ней указанной в задаче функции (задача 4.2.6).

Строки, структуры, связанные списки и файловый ввод-вывод на языке C++ (16 ч.)

12. Обработка символьных строк. Разработка *программы*, выполняющей чтения из указанного файла символьных строк и их обработку согласно условию (задача 5.1).
13. Создание, редактирование и файловый ввод-вывод линейного связанного списка структур. Разработка *программы*, осуществляющей: ввод с клавиатуры линейного связанного спи-

ска структур; вывод списка на экран; добавление новых структур в начало, в конец или в заданную позицию списка; удаление из списка структуры по ее номеру; сохранение в файле и чтение из файла всего списка структур; определение длины списка. Структуры из связного списка (элементы списка) должны содержать перечень своих элементов (полей), указанный в индивидуальном задании. Все перечисленные операции над списком и его элементами могут выбираться пользователем в произвольном порядке по номеру операции, вводимому им с клавиатуры. Каждая операция должна быть реализована в программе в виде отдельной функции (задача 5.2.2).

Источник задач:

Сборник задач по базовой компьютерной подготовке. / В. С. Зубов, И. Н. Котарова, О. Г. Батасова и др.; Составитель И. Н. Котарова. – М.: Изд-во МЭИ, 1998.