

Варіанти розрахунково-графічної роботи (C++)

Варіант 1. Розробити програму на C++ для обробки одновимірного масиву: знайти мінімальний і максимальний елементи та їх позиції.

Варіант 2. Реалізувати програму на C++ для сортування масиву методом вибору та порівняти кількість операцій із методом обміну.

Варіант 3. Розробити програму для пошуку елемента в масиві з використанням лінійного та бінарного пошуку.

Варіант 4. Створити програму на C++ для транспонування квадратної матриці розміру $N \times N$.

Варіант 5. Реалізувати алгоритм обчислення суми елементів головної та побічної діагоналей матриці.

Варіант 6. Розробити програму для обробки двовимірного масиву: визначити рядок з максимальною сумою елементів.

Варіант 7. Створити програму на C++ для роботи з рядками: підрахунок кількості слів у введеному тексті.

Варіант 8. Реалізувати програму для перевірки симетричності рядка (паліндром).

Варіант 9. Розробити програму для роботи з файлами: запис масиву чисел у файл та зчитування з файлу з подальшою обробкою.

Варіант 10. Створити програму на C++ для пошуку найчастіше зустрічаемого елемента масиву.

Варіант 11. Реалізувати рекурсивний алгоритм обчислення факторіала та порівняти його з ітеративним.

Варіант 12. Розробити рекурсивну програму для обчислення чисел Фібоначчі.

Варіант 13. Створити програму для видалення елемента з масиву за заданим індексом.

Варіант 14. Реалізувати програму на C++ для злиття двох відсортованих масивів в один відсортований.

Варіант 15. Розробити програму для обчислення середнього арифметичного, дисперсії та стандартного відхилення масиву.

Варіант 16. Створити програму для обробки текстового файлу: підрахунок кількості рядків, слів і символів.

Варіант 17. Реалізувати програму для пошуку найбільшого спільного дільника (НСД) двох чисел алгоритмом Евкліда.

Варіант 18. Розробити програму на C++ для перевірки правильності розміщення дужок у виразі.

Варіант 19. Створити програму для роботи зі структурами: зберігання та обробка даних про студентів.

Варіант 20. Реалізувати програму для обчислення та аналізу часової складності алгоритму сортування (експериментально).