Міністерство освіти і науки України Національний технічний університет України “Київський політехнічний інститут ім. Ігоря Сікорського” Факультет інформатики та обчислювальної техніки Кафедра автоматизованих систем обробки інформації та управління

ЗВІТ

про виконання лабораторного практикуму № 7 з теми:

“Одновимірні масиви”

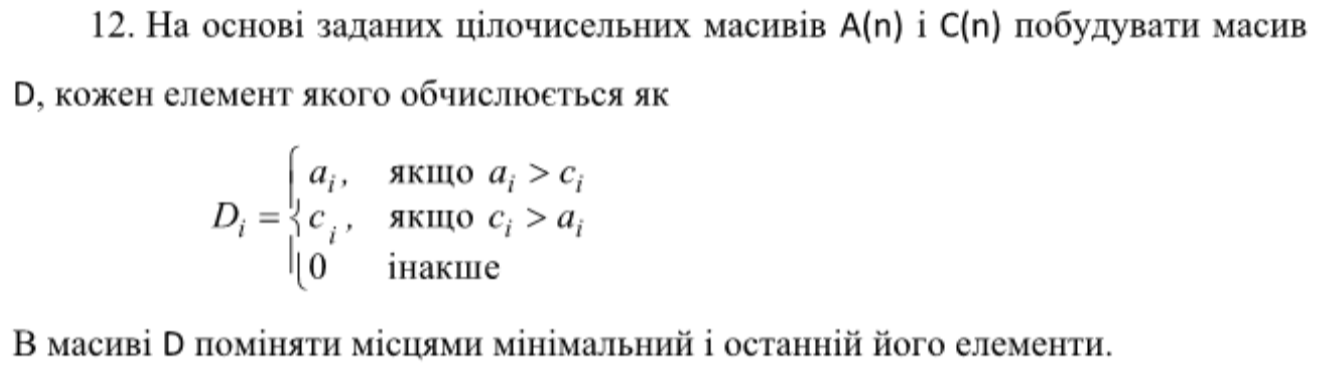
**Варіант № 12**

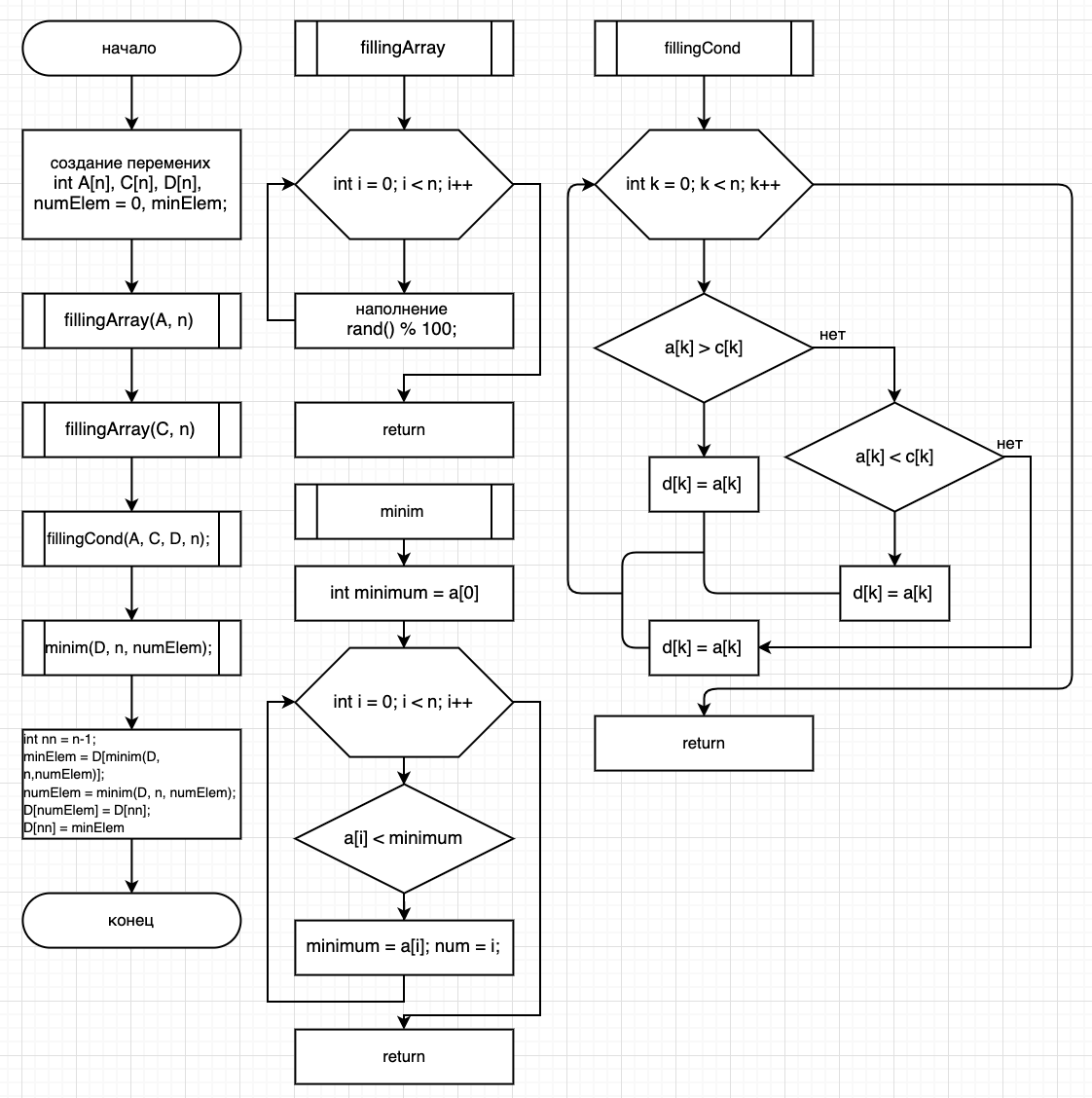
**Тема:** Одновимірні масиви

**Мета:** Вивчити особливості обробки одновимірні масивів

**Номер роботи та автор:** 12 варіант , Коробка Ілля Олександрович

**Умова задачі:**

****

**Блок схема:**

**Текст усіх файлів проекту: (main.cpp)**

**#include <iostream>**

**#include <ctime>**

**using namespace std;**

**int fillingArray(int a[], int n); // init funcs**

**int minim(int a[], int n, int num);**

**int fillingCond(int a[], int c[], int d[], int n);**

**int main(){**

**srand((unsigned int)time(NULL));**

**int n;**

**cout << "enter n: ";**

**cin >> n; // enter n**

**int A[n], C[n], D[n], numElem = 0, minElem; // creative variables**

**fillingArray(A, n); // call func for filling array**

**fillingArray(C, n);**

**fillingCond(A, C, D, n); // call func for filling array**

**minim(D, n, numElem); // call func for search min elem**

**// for (int i = 0; i < n; i++) {**

**// cout << D[i] << " / ";**

**// }**

**// cout << endl;**

**int nn = n-1; // change min elem and end elem**

**minElem = D[minim(D, n, numElem)];**

**numElem = minim(D, n, numElem);**

**D[numElem] = D[nn];**

**D[nn] = minElem;**

**// for (int i = 0; i < n; i++) {**

**// cout << D[i] << " / ";**

**// }**

**// cout << endl;**

**}**

**int fillingArray(int a[], int n){**

**for (int i = 0; i < n; i++){ // cycle for filling array random**

**a[i] = rand() % 100;**

**}**

**return a[n];**

**}**

**int minim(int a[], int n, int num){**

**int minimum = a[0];**

**for (int i = 0; i < n; i++) { // search min elem in cycle**

**if (a[i] < minimum) {**

**minimum = a[i];**

**num = i;**

**}**

**}**

**return num;**

**}**

**int fillingCond(int a[], int c[], int d[], int n){**

**for (int k = 0; k < n; k++) {**

**if (a[k] > c[k]) {**

**d[k] = a[k];**

**} else if (c[k] > a[k]) {**

**d[k] = c[k];**

**} else {**

**d[k] = 0;**

**}**

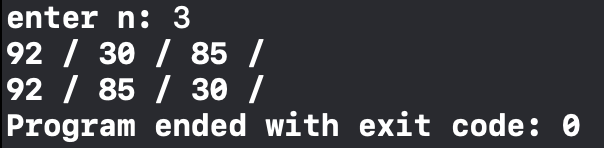
**}**

**return d[n];**

**};**

**Копии екранних форм результатів роботи з поясненням:(відео на GitHub)**

****

****

**Висновок:** Вивчили особливості обробки одновимірні масивів