

Міністерство освіти і науки України
Національний університет «Львівська політехніка»
Кафедра систем штучного інтелекту



Звіт

про виконання лабораторних та практичних робіт блоку № 4

На тему: «Одновимірні масиви. Двовимірні Масиви. Вказівники та Посилання. Динамічні масиви. Структури даних. Вкладені структури.

Алгоритми обробки та робота з масивами та структурами.»

з дисципліни: «Основи програмування»

до:

ВНС Лабораторної Роботи №4

ВНС Лабораторної Роботи №5

Алготестер Лабораторної Роботи №2

Алготестер Лабораторної Роботи №3

Практичних Робіт до блоку №4

Виконав(ла):

Студент групи ШІ-13

Басараб Дмитрій Богданович

Львів 2024

Тема роботи: Одновимірні масиви. Двовимірні Масиви. Вказівники та Посилання. Динамічні масиви. Структури даних. Вкладені структури. Алгоритми обробки та робота з масивами та структурами.

Мета роботи: Навчитись працювати з Одновимірними масивами, Двовимірними масивами, Вказівниками та Посиланнями, Динамічними масивами, Структурами даних та Вкладеними структурами. Зрозуміти Алгоритми обробки та робота з масивами та структурами.

Теоретичні відомості:

а.) C++
б.) Draw.io

а.) Лекції з ВНС, *GeekforGeeks*
б.) <https://yevshan.com.ua/info/006/content/content3.html>
GeekforGeeks

Порядок виконання роботи

Виконання роботи:

- 1) Опрацювання завдання та вимог до програм та середовища:
Завдання №1 - VNS Lab 4 var 20

20.

- 1) Реалізувати з використанням масиву двонаправлене кільце (перегляд можливий в обидва боки, від останнього елемента можна перейти до першого).
- 2) Роздрукувати отриманий масив, починаючи з К-ого елемента і до К-1 (по кільцю вліво).
- 3) Додати в кільце після елементів з індексами кратними 5 елементи, які дорівнюють 0.
- 4) Роздрукувати отриманий масив, починаючи з К-ого елемента (і до К+1 по кільцю вправо).

Завдання №2 - VNS Lab 5 var 20

20. Знайти мінімальний з не повторюваних елементів двовимірного масиву.

Завдання №3 - algotester Lab 2 var 1

Lab 2v1

Limits: 1 sec., 256 MiB

У вас є дорога, яка виглядає як N чисел.

Після того як ви по ній пройдете - вашу втому можна визначити як різницю максимального та мінімального елемента.

Ви хочете мінімізувати втому, але все що ви можете зробити - викинути одне число з дороги, тобто забрати його з масиву.

В результаті цієї дії, яку мінімальну втому ви можете отримати в кінці дороги?

Input

У першому рядку ціле число N - кількість чисел.

У другому рядку масив r , який складається з N цілих чисел.

Output

Єдине ціле число m - мінімальна втома, яку можна отримати.

Завдання №4 - algotester Lab 3 var 3

Lab 3v3

Limits: 1 sec., 256 MiB

Вам дана стрічка s .

Ваше завдання зробити компресію стрічки, тобто якщо якась буква йде більше одного разу підряд у стрічці замінити її на букву + кількість входжень підряд.

Input

У першому рядку стрічка S.

Output

Стрічка S compressed.

Завдання №5 - Practice work

Перевірка чи слово або число є паліндромом

Задача

Реалізувати програму, яка перевіряє, чи дане слово чи число є паліндромом за допомогою рекурсії.

Паліндром — це послідовність символів, яка однаково читається вперед і назад (наприклад, «radar», «level», «12321»).

Мета Задачі

Навчитися користуватися механізмами перевантаження функції та використовувати рекурсію для вирішення задач обчислення.

Вимоги:

1. Визначення функції:
 - a. Реалізуйте рекурсивну функцію *isPalindrome*, яка перевіряє, чи заданий рядок є паліндромом.
2. Приклад визначення функції:
 - a. *bool isPalindrome(const string& str, int start, int end);*
3. Перевантаження функцій:
 - a. Перевантажте функцію *isPalindrome* для роботи з цілими значеннями.
 - b. *bool isPalindrome(ціле число);*
4. Рекурсія:
 - a. Рекурсивна функція для рядків перевірить символи в поточній початковій і кінцевій позиціях. Якщо вони збігаються, він буде рекурсивно перевіряти наступні позиції, поки початок не перевищить кінець, після чого рядок буде визначено як паліндром.

Кроки реалізації

- Визначте та реалізуйте рекурсивну функцію isPalindrome для рядків.
- Визначте та реалізуйте перевантажену функцію isPalindrome для цілих чисел. Використати математичний підхід щоб перевірити чи число є паліндромом.

Завдання №6 - Self-algotester work (algotester lab3v2)

Lab 3v2

Limits: 1 sec., 256 MiB

Вам дано 2 масиви розміром N та M . Значення у цих масивах унікальні.

Ваше завдання вивести у першому рядку кількість елементів, які наявні в обох масивах одночасно, у другому кількість унікальних елементів в обох масивах разом.

Input

У першому рядку ціле число u

У другому рядку N цілих чисел $a_1..a_n$

У третьому рядку ціле число M

У четвертому рядку M цілих чисел $b_1..b_n$

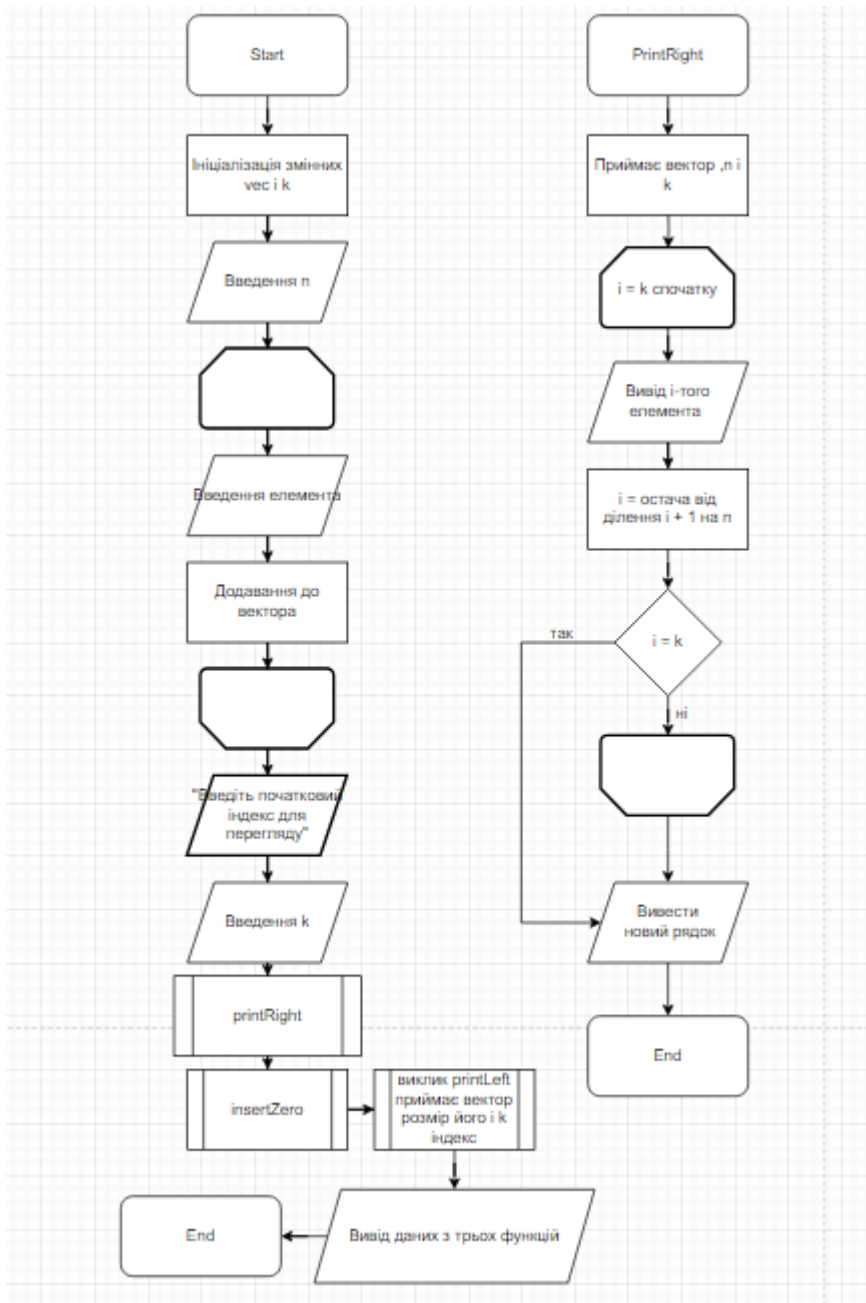
Output

У першому рядку одне ціле число - кількість елементів, які наявні в обох масивах одночасно.

У другому рядку кількість унікальних елементів в обох масивах (тобто кількість унікальних елементів у масиві, який буде об'єднанням двох даних).

2) Дизайн та планована оцінка часу виконання завдань:

Завдання №1 - lab 4 vns 20 variant

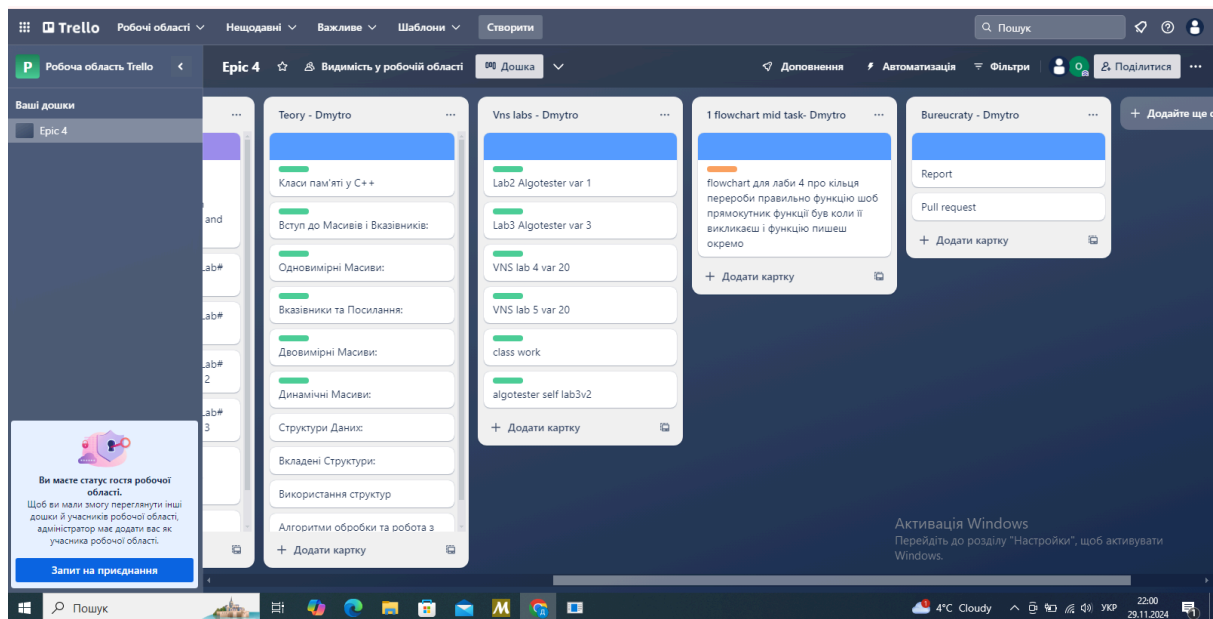




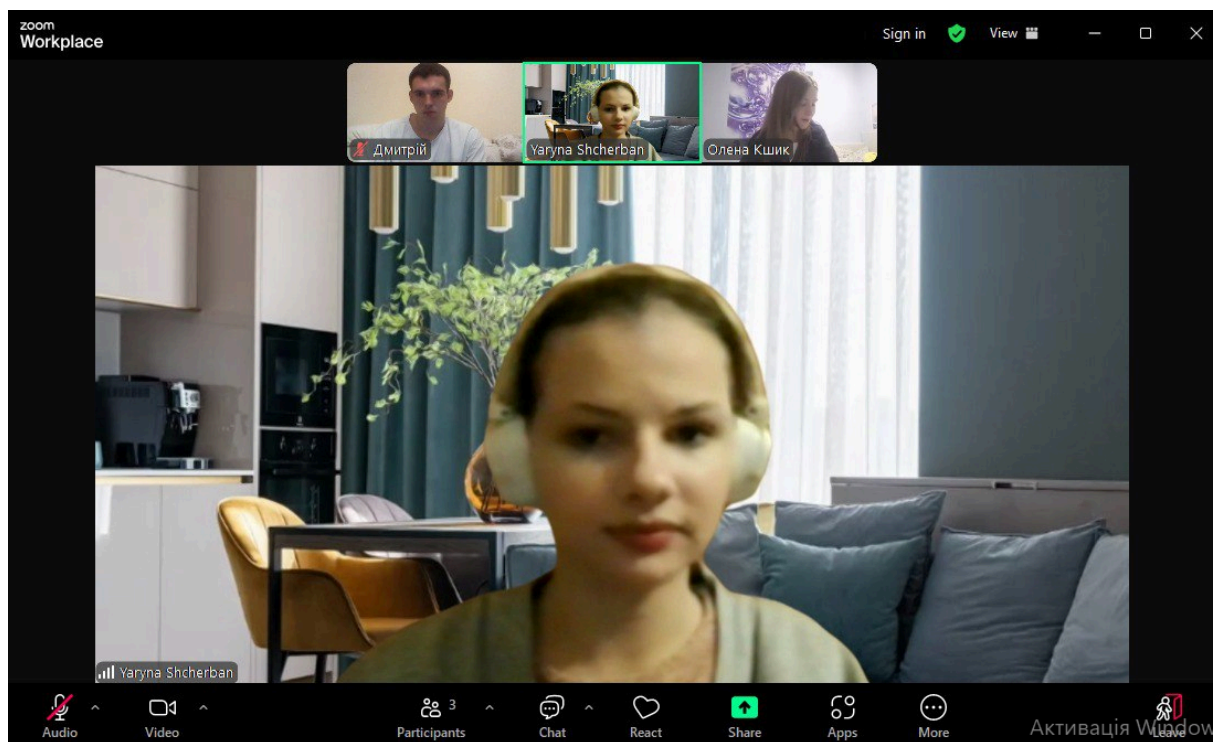
Планований час виконання завдання 1 год

3) Конфігурація середовища до виконання завдань:

Trello



Робота з командою



- 4) Код програм із задач:
Завдання №1 - VNS Lab 4 var 20


```

pointer.cpp  algotester_lab_2_variant_1_dmytrii_basarab.cpp  vns_lab_4_task_1_variant_20_dmytrii_basarab.cpp
epic_4 > vns_lab_4_task_1_variant_20_dmytrii_basarab.cpp > insertZeros(vector<int>&)
1  #include <iostream>
2  #include <vector>
3  using namespace std;
4
5  // Функція для роздрукування елементів кільця від певного індексу вправо
6  void printRight(vector<int>& vec, int n, int k) {
7      cout << "Перегляд вправо з індексу " << k << ": ";
8      for (int i = k; ; ) {
9          cout << vec[i] << " ";
10         i = (i + 1) % n; // Перехід вправо, циклічно від k до k
11         // 0 1 2 3 4   при i = 2 i n = 5 3/5 = 0 остача 3
12         //отже i стає 3 - наступним елементом
13         if (i == k) break; // Повертаємося до початкового індексу
14     }
15     cout << endl;
16 }
17
18 void insertZeros(vector<int>& vec) {
19     int i;
20     cout << "Нулі біля кратних 5: " << vec[i] << " ";
21     for (i = 0; i < vec.size(); ++i) {
22         if (i % 5 == 0) { //після елементів кратних 5 тому +1
23             vec.insert(vec.begin() + i + 1, 0);
24             i++; //пропускаєм доданий елемент
25         }
26         cout << vec[i] << " ";
27     }
28     cout << endl;
29 }
30
31 // Функція для роздрукування елементів кільця від певного індексу вліво
32 void printLeft(vector<int>& vec, int n, int k) {
33     cout << "Перегляд вліво з індексу " << k << ": ";
34     for (int i = k; ; ) {
35         cout << vec[i] << " ";
36         i = (i - 1 + n) % n; // Перехід вліво, циклічно, додаємо n щоб індекс не став від'ємним він переміщується в
37         if (i == k) break; // Повертаємося до початкового індексу
38     }
39     cout << endl;

```

Активуйте Windows
 Перейдіть до розділу "На
 Windows.

```

39 }
40
41 int main() {
42
43
44
45     vector<int> vec;
46     int n;
47     cout << "Розмір масиву: ";
48     cin >> n;
49     cout << "Елементи: ";
50     for (int i = 0; i < n; ++i) {
51         int q;
52         cin >> q;
53         vec.push_back(q);
54     }
55
56
57
58
59     int k;
60     cout << "Введіть початковий індекс для перегляду (від 0 до " << n-1 << "): ";
61     cin >> k;
62
63
64     printRight(vec, n, k);
65     insertZeros(vec);
66     printLeft(vec, n, k);
67
68
69     return 0;
70 }

```

Завдання №2 - VNS Lab 5 var 20

```
pointer.cpp vns_lab_5_task_1_variant_20_dmytrii_basarab.cpp X
epic_4 > vns_lab_5_task_1_variant_20_dmytrii_basarab.cpp > collect(int [100][100], int, int)
1  #include <iostream>
2  #include <vector>
3  #include <algorithm>
4
5  using namespace std;
6
7  // Функція для збору і знаходження унікальних значень
8  void collect(int arr[100][100], int rows, int cols) {
9      vector<int> v;
10     vector<int> res;
11
12     // Переносимо елементи з масиву в вектор
13     for (int i = 0; i < rows; i++) {
14         for (int j = 0; j < cols; j++) {
15             v.push_back(arr[i][j]);
16         }
17     }
18
19     sort(v.begin(), v.end());
20
21     // Знаходимо числа що не мають повторів
22     for (int l = 0; l < v.size(); l++) { //масив відсортований повторювані елементи один біля одного
23         if ((l == 0 || v[l] != v[l - 1]) && (l == v.size() - 1 || v[l] != v[l + 1])) {
24             res.push_back(v[l]);
25         }
26     }
27
28     if (res.empty()) {
29         cout << "Немає унікальних чисел!" << endl;
30     } else {
31         cout << "Мінімальне число без повторів: " << *min_element(res.begin(), res.end()) << endl;
32     }
33 }
34
35 int main() {
36     int rows, cols;
37
38     cout << "Введіть кількість рядків: ";
39     cin >> rows;
40     cout << "Введіть кількість стовпців: ";
41     cin >> cols;
42
43     int arr[100][100];
44
45     cout << "Введіть елементи масиву:" << endl;
46     for (int i = 0; i < rows; i++) {
47         for (int j = 0; j < cols; j++) {
48             cin >> arr[i][j];
49         }
50     }
51
52     // Викликаємо функцію для знаходження унікальних чисел
53     collect(arr, rows, cols);
54
55     return 0;
56 }
57
58
59
60
```

Завдання №3 - algotester Lab 2 var 1

```
pointer.cpp  algotester_lab_2_variant_1_dmytrii_basarab.cpp X
epic_4 > G algotester_lab_2_variant_1_dmytrii_basarab.cpp > main()
1  #include <iostream>
2  #include <algorithm>
3  int k = 0;
4  using namespace std;
5  int main() {
6      int N;
7      cin >> N;
8      if (N<=0) return 0;
9
10     int* r = new int[N];
11     for (int i=0; i < N; i++) {
12         cin >> r[i];
13         if(r[i]<0) {
14             return 0;
15         }
16     }
17
18     for (int k = 0;k<N;k++) {
19         for (int j=k+1;j<N;j++) {
20             if(r[k]>r[j]) {
21                 swap(r[k],r[j]);
22             }
23         }
24     }
25     if(N==1){
26         cout << r[0] - r[0] << endl;
27         return 0;
28     }
29     int m1;
30     int m2;
31     int* ptr1 = &m1; //показав роботу і синтаксис вказівників
32     int* ptr2 = &m2;
33     *ptr1 = r[N - 2] - r[0];
34     *ptr2 = r[N - 1] - r[1];
35     if (*ptr1<*ptr2) {
36         cout << *ptr1;
37     } else cout << *ptr2;
38     delete[] r;
39     return 0;
40 }
```

Завдання №4 - algotester Lab 3 var 3

```
pointer.cpp  algotester_lab_3_task_dmytrii_basarab.cpp X
epic_4 > algotester_lab_3_task_dmytrii_basarab.cpp > main()
1  #include <iostream>
2  #include <string>
3  #include <algorithm>
4
5  using namespace std;
6  int main() {
7      string res("");
8      string S;
9      cin >> S;
10     int num = 1;
11
12     for (int i=1;i<=S.length();i++) {
13         if(S[i] == S[i-1]) {
14             num++;
15         }else /*if (S[i] == S[i-1] && S[i]!=S[S.length() - 1])*/ { //ми зара на b
16             res+=S[i-1];
17             if (num > 1) {
18                 res += to_string(num);
19                 num = 1;
20                 //} все було добре для двох до оцєї знизу умови
21                 if ( S[i]==S[S.length() - 1]) {
22                     num = 1;
23                 }
24             }
25         }
26     }
27 }
28
29
30 cout << res;
31 return 0;
32 }
33
```

Завдання №5 - Practice work

```
pointer.cpp  practice_work_self_algotester_tasks_dmytrii_basarab.cpp  practice_work_team_tasks_dmytrii_basarab.cpp
epic_4 > practice_work_team_tasks_dmytrii_basarab.cpp > isPalindrome(const string &, int, int)
1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  bool isPalindrome(const string& word, int start, int end){
6      if (start > end) return true;
7      if (word[start] != word[end]) return false;
8      return isPalindrome(word, start+1, end-1);
9  }
10 bool isPalindrome(int& num) {
11     if (num < 0) {
12         return false;
13     }
14     int a = num;
15     int reversed = 0;
16     while (num > 0) {
17         int digit = num % 10;
18         num = num / 10;
19         reversed = reversed * 10 + digit;
20     }
21     return reversed == a;
22 }
23 int main() {
24     string word;
25     cin >> word;
26     if (isPalindrome(word, 0 , word.length() - 1)) {
27         cout << "yes" << endl;
28     } else {
29         cout << "no" << endl;
30     }
31
32     int num;
33     cin >> num;
34     if (isPalindrome(num)) {
35         cout << "yes" << endl;
36     } else {
37         cout << "no" << endl;
38     }
39     return 0;
}
```

Завдання №6 - Self-algotester work (algotester lab3v2)

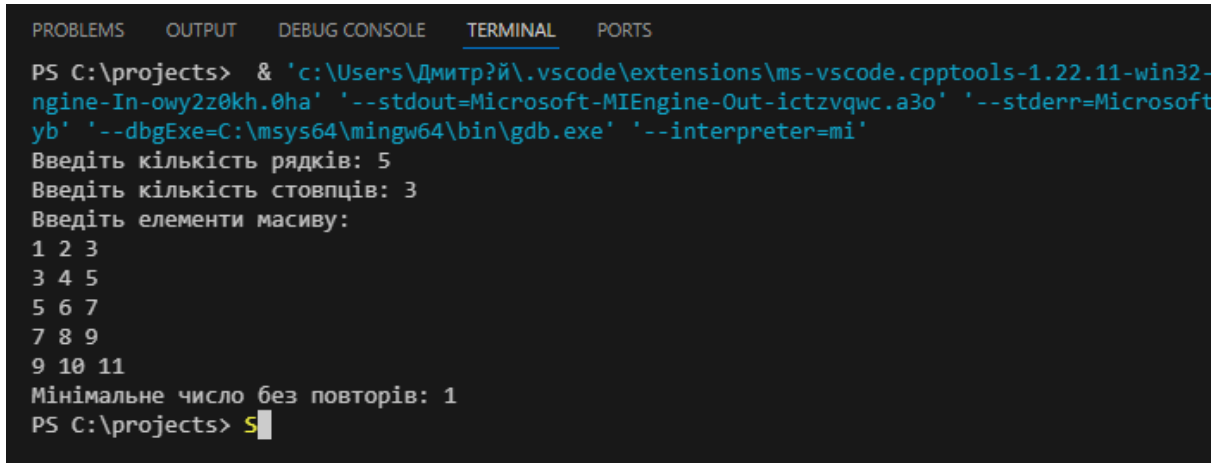
```
pointer.cpp  practice_work_self_algotester_tasks_dmytrii_basarab.cpp
epic_4 > practice_work_self_algotester_tasks_dmytrii_basarab.cpp > main()
1  #include <iostream>
2
3  using namespace std;
4
5  int main() {
6      int common = 0;
7      int unique = 0;
8
9      struct dynamic {
10         int M;
11     } forarr1, forarr2;
12
13     cin >> forarr1.M;
14     int* arr1 = new int[forarr1.M];
15     for (int i=0; i<forarr1.M; i++) {
16         cin >> arr1[i];
17     }
18
19     cin >> forarr2.M;
20     int* arr2 = new int[forarr2.M];
21     for (int j=0; j<forarr2.M; j++) {
22         cin >> arr2[j];
23     }
24
25     for (int k = 0; k < forarr1.M; k++){
26         for (int q = 0; q < forarr2.M; q++){
27             if (arr1[k]==arr2[q]) {
28                 common++;
29             }
30         }
31     }
32
33     unique = forarr1.M+forarr2.M-common;
34     cout << common << endl;
35     cout << unique << endl;
36     delete[] arr1;
37     delete[] arr2;
38     return 0;
39 }
```

5) *Результати виконання завдань, тестування та фактично затрачений час:*
Завдання №1 - VNS Lab 4 var 20

```
PS C:\projects>
PS C:\projects> & 'c:\Users\?\.vscode\extensions\ms-vscode.cpptools-1.22.11-win32-x64\debugAdapters\bin\In-dbg\jvpmw.jwi' '--stdout=Microsoft-MIEngine-Out-33d323rc.qaz' '--stderr=Microsoft-MIEngine-Error-hvspmfg' -dbgExe=C:\msys64\mingw64\bin\gdb.exe' '--interpreter=mi'
Розмір масиву: 5
Елементи: 1 2 3 4 5
Введіть початковий індекс для перегляду (від 0 до 4): 2
Перегляд вправо з індексу 2: 3 4 5 1 2
Нулі біля кратних 5: 1 0 2 3 4 0
Перегляд вліво з індексу 2: 2 0 1 4 3
PS C:\projects>
```

Фактично витрачений час 2 год

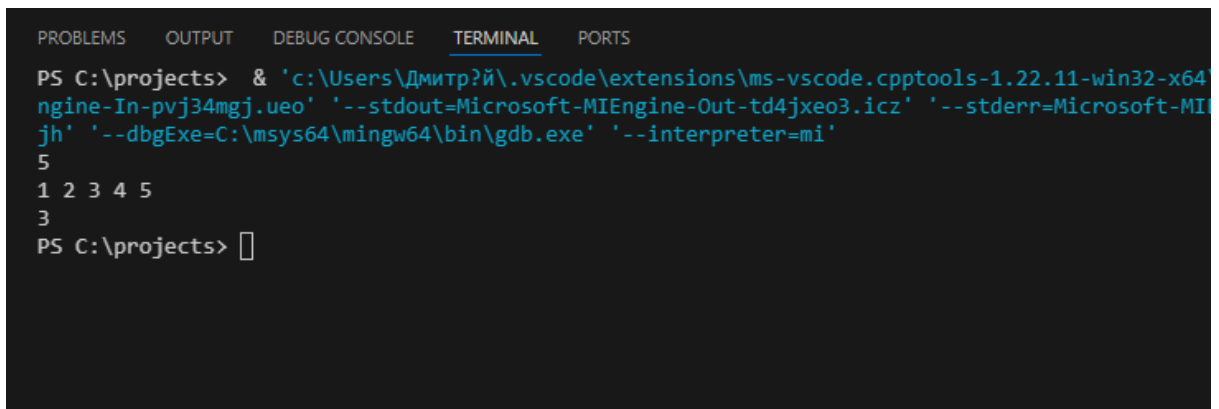
Завдання №2 - VNS Lab 5 var 20



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\projects> & 'c:\Users\Дмитр?й\.vscode\extensions\ms-vscode.cpptools-1.22.11-win32-
ngine-In-owy2z0kh.0ha' '--stdout=Microsoft-MIEngine-Out-ictzvqwc.a3o' '--stderr=Microsoft
yb' '--dbgExe=C:\msys64\mingw64\bin\gdb.exe' '--interpreter=mi'
Введіть кількість рядків: 5
Введіть кількість стовпців: 3
Введіть елементи масиву:
1 2 3
3 4 5
5 6 7
7 8 9
9 10 11
Мінімальне число без повторів: 1
PS C:\projects> S
```

Фактично витрачений час 1 год

Завдання №3 - algotester Lab 2 var 1



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\projects> & 'c:\Users\Дмитр?й\.vscode\extensions\ms-vscode.cpptools-1.22.11-win32-x64-
ngine-In-pvj34mgj.ueo' '--stdout=Microsoft-MIEngine-Out-td4jxeo3.icz' '--stderr=Microsoft-MI
jh' '--dbgExe=C:\msys64\mingw64\bin\gdb.exe' '--interpreter=mi'
5
1 2 3 4 5
3
PS C:\projects> 
```

Фактично витрачений час 30 хв

Завдання №4 - algotester Lab 3 var 3


```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS C:\projects> & 'c:\Users\Дмитр?й\.vscode\extensions\ms-vscode.cpptools-1.22.11-
engine-In-cpzcuyff.21b' '--stdout=Microsoft-MIEngine-Out-r04gnyah.u1u' '--stderr=Mic
wf' '--dbgExe=C:\msys64\mingw64\bin\gdb.exe' '--interpreter=mi'
AAAAabbcdddd
A2a3b2cd5
PS C:\projects> █
```

Фактично витрачений час 40 хв

Завдання №5 - Practice work

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS C:\projects> & 'c:\Users\Дмитр?й\.vscode\extensions\ms-vscode.cpptools-1.22.11-
engine-In-hltrlic4.ozu' '--stdout=Microsoft-MIEngine-Out-tb1o4cdj.r4h' '--stderr=Mic
lb' '--dbgExe=C:\msys64\mingw64\bin\gdb.exe' '--interpreter=mi'
level
yes
12341
no
PS C:\projects> █
```

Фактично витрачений час 2 год

Завдання №6 - Self-algotester work (algotester lab3v2)

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

engine-In-u5fljvmt.com' '--stdout=Microsoft-MIEngine-Out-p4xcbr5m.rkp' '--std
bs' '--dbgExe=C:\msys64\mingw64\bin\gdb.exe' '--interpreter=mi'
5
1 2 3 4 5
5
5
4 5 6 7 8
2
8
PS C:\projects> ^C
PS C:\projects>
PS C:\projects> & 'c:\Users\?\vscode\extensions\ms-vscode.cpptools-1.22.11
In-rhdlaqb.vww' '--stdout=Microsoft-MIEngine-Out-14f12btk.m1c' '--stderr=Mi
-dbgExe=C:\msys64\mingw64\bin\gdb.exe' '--interpreter=mi'
5
1 2 3 4 5
5
3 4 5 6 7
3
7
PS C:\projects> █
```

Фактично витрачений час 50 хв

Висновки: На цій лабораторній роботі я навчився працювати з Одновимірними масивами, Двовимірними масивами, Вказівниками та Посиланнями, Динамічними масивами, Структурами даних та Вкладеними структурами.Зрозуміти Алгоритми обробки та робота з масивами та структурами.