Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра систем штучного інтелекту



**Звіт**

про виконання

**Лабораторних та практичних робіт №3**

***з дисципліни:*** «Мови та парадигми програмування»

***з розділу***: «Прості структури даних. Одновимірні масиви. Двовимірні масиви. Алгоритми обробки.»

***Виконала:***

студентка групи ШІ-13

Кузнєцова Анастасія Володимирівна

# **Тема роботи:**

Робота з одновимірними масивами. Вивчення одновимірних та двовимірних масивів. Оптимізація обробки масивів та вивчення простих структур даних. Опрацювання алгоритмів обробки.

# **Мета роботи:**

Отримання навичок ефективної роботи з одновимірними масивами та їх оптимізація. Вивчення алгоритмів обробки масивів та їх застосування для ефективного програмування. Освоєння основ простих структур даних та їх вплив на роботу з масивами.

# **Теоретичні відомості:**

1. Теоретичні відомості з переліком важливих тем:

* Тема №1: Одновимірні масиви.
* Тема №2: Двовимірні масиви.
* Тема №3: Генерація випадкових чисел.
* Тема №4: Перетворення числа у string.

1. Індивідуальний план опрацювання теорії:

* Тема №1: Одновимірні масиви.
  + Джерела Інформації:
    - Стаття: https://www.bestprog.net/uk/2017/03/01/%D0%BC%D0%B0%D1%81%D0%B8%D0%B2%D0%B8-%D1%87%D0%B0%D1%81%D1%82%D0%B8%D0%BD%D0%B0-1-%D0%B2%D0%B8%D0%B7%D0%BD%D0%B0%D1%87%D0%B5%D0%BD%D0%BD%D1%8F-%D0%BC%D0%B0%D1%81%D0%B8%D0%B2%D1%83-%D0%BE%D0%B4/  
      https://cherto4ka.xyz/2020/01/16/massive\_begin/  
      https://acode.com.ua/urok-78-fiksovani-masyvy/
    - Відео: https://www.youtube.com/watch?v=ULdbOaMBPYc
  + Що опрацьовано: Використання одновимірних масивів, їх особливості. Оголошення одновимірних масивів. Надання значень елементам масиву.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 22.11.23
  + Звершення опрацювання теми: 01.12.23
* Тема №2: Двовимірні масиви.
  + Джерела Інформації:
    - Відео: https://www.youtube.com/watch?v=V2g3B9Zbh4Q
    - Стаття: https://acode.com.ua/urok-81-bagatovymirni-masyvy/
  + Що опрацьовано:
    - Використання двовимірних масивів. Оголошення двовимірного масиву. Надання значень елементам масиву.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 26.11.23
  + Звершення опрацювання теми: 01.12.23
* Тема №3: Генерація випадкових чисел.
  + Джерела Інформації:
    - Стаття: https://acode.com.ua/urok-74-generatsiya-randomnyh-chysel/  
      https://www.bestprog.net/uk/2020/08/13/%D1%81-generating-random-numbers-functions-rand-srand-time-examples-ua/
  + Що опрацьовано:  
    Як створити генератор випадкових чисел. Використання функцій rand(), srand(), time().
  + Статус: Ознайомлений
  + Початок опрацювання теми: 27.11.23
  + Звершення опрацювання теми: 30.12.23
* Тема №4: Перетворення числа у string.
  + Джерела Інформації:
    - Стаття: https://www.geeksforgeeks.org/converting-number-to-string-in-cpp/  
      https://acode.com.ua/urok-60-vvedennya-v-std-string/
  + Що опрацьовано:
    - Перетворення чисел типу int, long та інші у тип string.
  + Статус: Ознайомлений
  + Початок опрацювання теми: 28.11.23
  + Звершення опрацювання теми: 29.11.23

# **Виконання роботи:**

## **1. Опрацювання завдання та вимог до програм та середовища:**

Завдання №1 **VNS Lab 4**

* Варіант завдання: 8
* Деталі завдання: Сформувати одновимірний масив цілих чисел, використовуючи генератор випадкових чисел. Роздрукувати отриманий масив, знищити всі елементи із заданим значенням. Додати перед кожним парним елементом масиву елемент зі значенням 0, та роздрукувати отриманий масив.
* Важливі деталі для врахування в імплементації програми: Для виконання роботи використовувати статичні масиви із псевдозмінними межами, оголосити масив достатньої величини. Потім користувач вводить реальну довжину масиву. При зменшенні або збільшенні довжини масиву необхідно змінювати його реальну довжину.

Завдання №2 **VNS Lab 5**

* Варіант завдання: 8
* Деталі завдання: Написати функцію транспонування квадратної матриці та визначити за її допомогою, чи є задана матриця симетричною.
* Важливі деталі для врахування в імплементації програми: Матриця вважається симетричною, якщо транспонована матриця дорівнює початковій

Завдання №3 **Algotester Lab 2**

* Варіант завдання: 3  
  Деталі завдання: Масив цілих чисел розміром N містить інформацію про розміщення двох дронів на першій та останній клітинці, які одночасно взлітають. На початку кожного ходу швидкість дрону стає рівною значенню клітинки, у якій він знаходиться. Коли обидва дрони опиняються в одній клітинці, виводиться "Collision". Якщо лівий дрон опиниться справа від правого, виводиться "Miss". У випадку, коли вони зупиняються один напроти одного виводиться "Stopped".
* Важливі деталі для врахування в імплементації програми: Перевірка виконується також до взльоту. Дрони не можуть вилетiти за межi [1, N] та їх швидкiсть завжди бiльша за нуль.

Завдання №4 **Algotester Lab 3**

* Варіант завдання: 3
* Деталі завдання: Зробити компресiю стрiчки, тобто якщо якась буква йде бiльше одного разу пiдряд у стрiчцi замiнити її на букву + кiлькiсть входжень пiдряд.
* Важливі деталі для врахування в імплементації програми: Великi та маленькi букви вважаються рiзними.

Завдання №5: **Class Practice Work**

* Деталі завдання: Реалізувати програму, щоб перевірити, чи є дане слово чи число паліндромом за допомогою рекурсії. Визначити та реалізувати рекурсивну функцію isPalindrome для рядків. Також визначити та реалізувати перевантажену функцію isPalindrome для цілих чисел. Для цього перетворити ціле число на рядок і використати рядкову версію функції для перевірки, чи є це паліндром.
* Важливі деталі для врахування в імплементації програми: Написати окремо функцію для визначення чи слово паліндром та функцію для перетворення числа у стрічку і подальшого виклику попередньої функції.

## **2. Дизайн та планована оцінка часу виконання завдань:**

Програма №1 **VNS Lab 4**

* Планований час на реалізацію: 1 день
* Важливі деталі для врахування в імплементації: Для виконання роботи використовувати статичні масиви із псевдозмінними межами, оголосити масив достатньої величини. Потім користувач вводить реальну довжину масиву. При зменшенні або збільшенні довжини масиву необхідно змінювати його реальну довжину.

Програма №2 **VNS Lab 5**

* Планований час на реалізацію: 4 години
* Важливі деталі для врахування в імплементації: Матриця вважається симетричною, якщо транспонована матриця дорівнює початковій

Програма №3 **Algotester Lab 2**

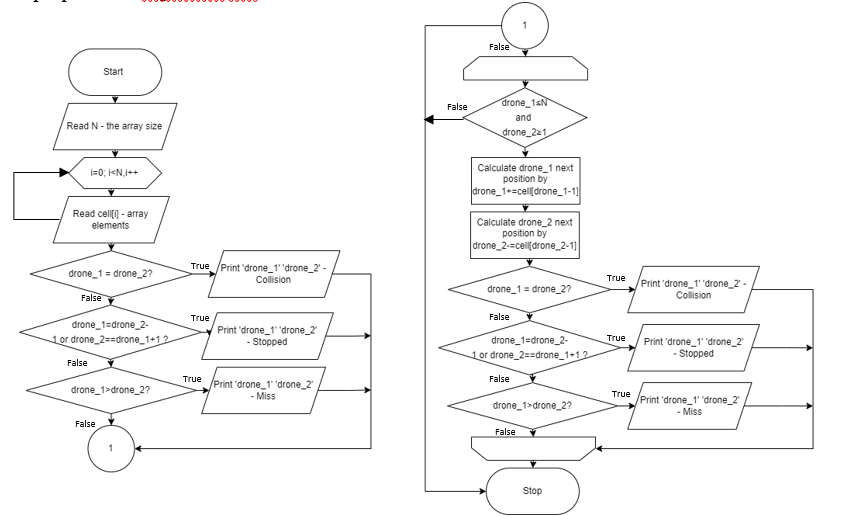


Рисунок 1 Блок-схема до Algotester Lab 2

* Планований час на реалізацію: 4 години
* Важливі деталі для врахування в імплементації: Перевірка виконується також до взльоту. Дрони не можуть вилетiти за межi [1, N] та їх швидкiсть завжди бiльша за нуль.

Програма №4 **Algotester Lab 3**

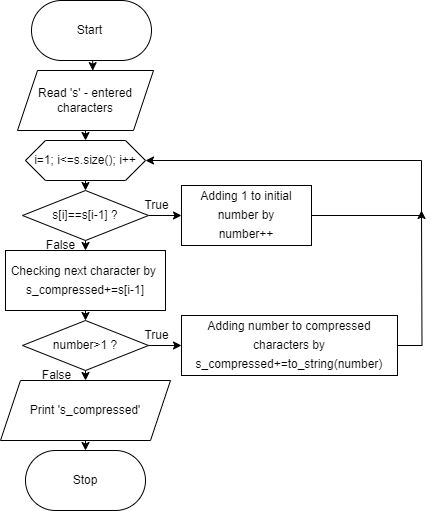


Рисунок 2 Блок-схема до Algotester Lab 3

* Планований час на реалізацію: 2 години
* Важливі деталі для врахування в імплементації: Великi та маленькi букви вважаються рiзними.

Програма №5 **Class Practice Work**

* Планований час на реалізацію: 6 годин
* Важливі деталі для врахування в імплементації: Написати окремо функцію для визначення чи слово паліндром та функцію для перетворення числа у стрічку і подальшого виклику попередньої функції.

## **3. Конфігурація середовища до виконання завдань:**

Файл settings.json в якому зберігаються налаштування VS Code

{

*"workbench.colorTheme"*: "Monokai",

*"code-runner.runInTerminal"*: true,

*"C\_Cpp.default.compilerPath"*: "d:\\C C++\\test.c",

*"cmake.configureOnOpen"*: true,

*"terminal.integrated.defaultProfile.windows"*: "Windows PowerShell",

*"terminal.integrated.profiles.windows"*: {

*"PowerShell"*: {

*"source"*: "PowerShell",

*"icon"*: "terminal-powershell"

        },

*"Command Prompt"*: {

*"path"*: [

                "${env:windir}\\Sysnative\\cmd.exe",

                "${env:windir}\\System32\\cmd.exe"

            ],

*"args"*: [],

*"icon"*: "terminal-cmd"

        },

*"Git Bash"*: {

*"source"*: "Git Bash"

        },

*"bash (MSYS2)"*: {

*"path"*: "C:\\msys64\\usr\\bin\\bash.exe",

*"args"*: [

                "--login",

                "-i"

            ]

        },

*"Windows PowerShell"*: {

*"path"*: "C:\\Windows\\System32\\WindowsPowerShell\\v1.0\\powershell.exe"

        }

    },

*"workbench.colorCustomizations"*: {}

}

## **4. Код програм з посиланням на зовнішні ресурси:**

Завдання №1 **VNS Lab 4**

#include <iostream>

#include <cstdlib>

#include <ctime>

using *namespace* std;

*int* N=100, n=0;

*int* main()

{

*int* a[N], value;

    cout<<"Enter the length of the array: ";

    cin>>N;

    srand(static\_cast<*unsigned* *int*>(time(0)));

    for (*int* i=0; i<N; i++)

    {

        a[i]=rand()%100;

        cout<<a[i]<<" ";

    }

    cout<<"\nEnter a value you want to delete: ";

    cin>>value;

    for (*int* i=0; i<N; i++)

    {

        if (a[i]!=value)

        {

            a[n]=a[i];

            n++;

        }

    }

    for (*int* i=n-1; i>0; i--)

    {

        if (a[i]%2==0)

        {

            for (*int* j = n; j > i; j--)

            {

                a[j] = a[j - 1];

            }

            a[i] = 0;

            n++;

        }

    }

    for (*int* i=0; i<n; i++)

    {

        cout<<a[i]<<" ";

    }

return 0;

}

Програма прописує одновимірний масив цілих чисел, які є не більше 99, використовуючи генератор випадкових чисел, та роздруковує його. Наступним кроком вона видаляє зазначений користувачем елемент. Далі, перед кожним парним елементом масиву, програма додає елемент зі значенням 0. В кінці програма роздруковує отриманий масив.

Pull request: https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/623/files#diff-ecb778c453e3a68209ad106580775f25b40bdbf824b5ff7103d38170a79cf61e

Завдання №2 **VNS Lab 5**

#include <iostream>

using *namespace* std;

#define N 100

*void* trans(*int* *A*[N][N], *int* *AT*[N][N], *int* &*n*)

{

    cout<<"Enter the matrix size: ";

    cin>>*n*;

    for (*int* i=0; i<*n*; i++)

    {

        for (*int* j=0; j<*n*; j++)

        {

            cout<<"Enter the matrix element: ";

            cin>>*A*[i][j];

        }

    }

    for (*int* i=0; i<*n*; i++)

    {

        for (*int* j=0; j<*n*; j++)

        {

            cout<<" "<<*A*[i][j];

            if (j == *n*-1)

            {

                cout<<"\n";

            }

        }

    }

    for (*int* i=0; i<*n*; i++)

    {

      for (*int* j=0; j<*n*; j++)

        {

*AT*[j][i] = *A*[i][j];

        }

    }

    cout<<"\n";

    for (*int* i=0; i<*n*; i++)

    {

        for (*int* j=0; j<*n*; j++)

        {

            cout<<" "<<*AT*[i][j];

            if (j == *n*-1)

            {

                cout<<"\n";

            }

        }

    }

}

*int* symetric(*int* *A*[N][N], *int* *AT*[N][N], *int* *n*)

{

    for (*int* i=0; i<*n*; i++)

    {

        for (*int* j=0; j<*n*; j++)

        {

            if (*A*[i][j]!=*AT*[i][j])

            {

                return false;

            }

        }

    }

    return true;

}

*int* main ()

{

*int* A[N][N], AT[N][N], n;

    trans(A, AT, n);

    if (symetric(A, AT, n))

    {

        cout<<"The matrix is symetric";

    }

    else

    {

        cout<<"The matrix isn't symetric";

    }

return 0;

}

Програма спочатку викликає функцію яка не повертає значення, де просить ввести розмір матриці n\*n (квадратна матриця), потім ввести елементи матриці (спочатку рядки, потім стовпці). Коли доходить до елемента, меншого ніж розмір матриці, переходить на новий рядок. Далі транспонує отриману матрицю, та друкує її за таким самим принципом, як і початкову. Потім програма викликає функцію, для визначення чи матриця є симетричною. Якщо початкова матриця не дорівнює транспонованій, то видає false. Останнім програма викликає функцію main в якій викликаються дві попередні функції і якщо матриця симетрична виводить це в термінал, якщо ні, то виводить, що вона є не симетричною.

Pull request: https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/623/files#diff-4a71a768d669ec2f4a4dcf457542eb219b8cd20c6cb3b63adaf06c8855dd6c41

Завдання №3 **Algotester Lab 2**

#include <iostream>

using *namespace* std;

*int* main()

{

*int* N;

cin>>N;//Введіть розмір масиву

*int* cell[N];

*int* drone\_1=1, drone\_2=N;

    for(*int* i=0; i<N; i++)

    {

        cin>>cell[i];//Введіть елементи масиву

    }

    if(drone\_1 == drone\_2)

    {

        cout<<drone\_1<<" "<<drone\_2<<"\n"<<"Collision";

        return 0;

    }

    else if (drone\_1==drone\_2-1 || drone\_2==drone\_1+1)

        {

            cout<<drone\_1<<" "<<drone\_2<<"\n"<<"Stopped";

            return 0;

        }

    else if (drone\_1>drone\_2)

        {

            cout<<drone\_1<<" "<<drone\_2<<"\n"<<"Miss";

            return 0;

        }

    else

    {

        while (drone\_1<=N && drone\_2>=1)

        {

            drone\_1+=cell[drone\_1-1];

            drone\_2-=cell[drone\_2-1];

            if(drone\_1 == drone\_2)

            {

                cout<<drone\_1<<" "<<drone\_2<<"\n"<<"Collision";

                break;

            }

            else if (drone\_1==drone\_2-1 || drone\_2==drone\_1+1)

            {

                cout<<drone\_1<<" "<<drone\_2<<"\n"<<"Stopped";

                break;

            }

            else if (drone\_1>drone\_2)

            {

                cout<<drone\_1<<" "<<drone\_2<<"\n"<<"Miss";

                break;

            }

        }

    }

return 0;

}

Завдання полягало в тому, щоб вивести фiнальну позицiю першого та другого дрона та одне зі слів: Collision, Miss, Stopped. Програма просить ввести користувача цiле число n - розмiр масиву, та елементи масиву. Перший дрон поміщується на 1 позицію, другий – n. Якщо на початку дрони на одній клітинці, то виводиться «Collision», якщо на клітинках поряд - «Stopped», якщо перший дрон на клітинках після другого - «Miss». З кожним випадком, в рядку перед текстом виводиться значення на яких розміщуються дрони. Якщо ж жодна з цих умов не виконалась далі програма прораховує переміщення дронів. Умови виводу повторюються як і спочатку.

Pull request: https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/623/files#diff-fe64a75aa2de00e92987f0d9daeee0be06bfe756c7ff8af8c907c3f4cabeafd1

Завдання №4 **Algotester Lab 3**

#include <iostream>

#include <string>

using *namespace* std;

*int* main ()

{

    string s, s\_compressed;

    cin>>s;

*int* number=1;

    for (*int* i=1; i<=s.size(); i++)

    {

       if (s[i]==s[i-1])

        {

            number++;

        }

        else

        {

            s\_compressed+=s[i-1];

            if(number>1)

            {

            s\_compressed+=to\_string(number);

            }

            number=1;

        }

    }

    cout<<s\_compressed;

    return 0;

}

Програма виконує компресію стрічки, яку ввів користувач. Якщо якась буква повторюється більше одного разу підряд у стрічці, то вона замінюється на букву + кількість входжень підряд.   
Pull request: https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/623/files#diff-19333dcec2e5c27b2eaa3a5402c8763b60c4f77bbdbfaad0981e2f4aba475d42

Завдання №5 **Class Practice Work**

#include <iostream>

#include <string>

using *namespace* std;

*bool* isPalindrome(string& *str*, *int* *start*, *int* *end*)

{

    if (*start*>*end*)

        return 0;

    if (*start*==*end*)

        return true;

return isPalindrome(*str*, *start*+1, *end*-1);

}

*bool* isPalindrome(*int* *number*)

{

    string str=to\_string(*number*);

return isPalindrome(str, 0, str.size()-1);

}

*int* main()

{

    string str = "level";

    if (isPalindrome(str, 0, str.size()-1))

    {

        cout<<"\nThe word is palindrome";

    }

    else

    {

        cout<<"\nThe word isn't palindrome";

    }

*int* number = 263962;

    if (isPalindrome(number))

    {

        cout<<"\nThe number is palindrome";

    }

    else

    {

        cout<<"\nThe number isn't palindrome";

    }

    return 0;

}

Програма реалізує перевірку, чи є введене слово чи число паліндромом за допомогою рекурсії. Для цього визначено рекурсивну функцію isPalindrome, яка приймає рядки і перевіряє, чи вони є паліндромами. Також визначена перевантажена функція isPalindrome для цілих чисел, яка перетворює число у рядок і використовує раніше визначену функцію для перевірки паліндромності. Програма використовує ці функції для перевірки введених даних і повідомляє, чи є вони паліндромами чи ні.

Pull request: https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/623/files#diff-5291e3d8e4be8bae58c63a21274a4135c9753c20f8eea80a9eb76e1d320154f0

## **5. Результати виконання завдань, тестування та фактично затрачений час:**

Завдання №1 **VNS Lab 4**

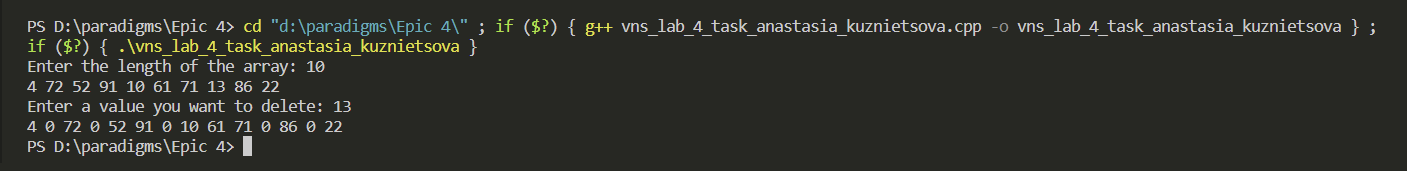


Рисунок 3 Виконання VNS Lab 4

Введений розмір масиву 10, відповідно програма виводить 10 випадкових значень. Далі вона просить обрати елемент, який користувач хоче видалити – 13. Програма видаляє елемент «13». Далі вона рахує частку від ділення кожного виведеного числа на 2. Якщо вона є, то програма пропускає це число, якщо її немає, то ставить перед ним 0.

Час, затрачений на виконання завдання: 1 день

Завдання №2 **VNS Lab 5**

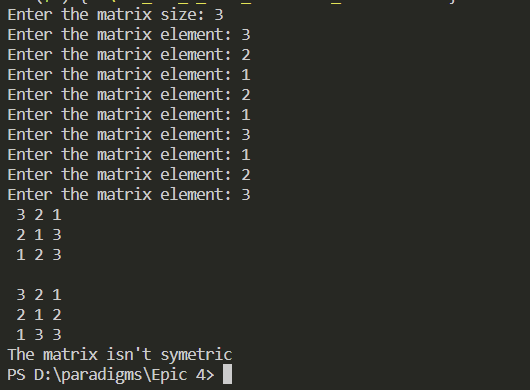


Рисунок 4 Виконання VNS Lab 5

Програма просить ввести розмір матриці – 3. Відповідно до розміру вона просить ввести елементи масиву. Далі рахує та виводить транспоновану матрицю і так як елементи розміщені не симетрично, тобто транспонована матриця не дорівнює початковій, виводить що матриця не є симетричною.

Час, затрачений на виконання завдання: 1 день

Завдання №3 **Algotester Lab 2**

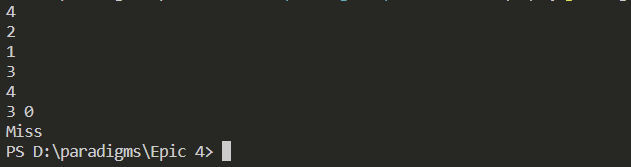


Рисунок 5 Виконання Algotester 2

Спочатку користувач вводить розмір масиву – 4. Відповідно до цього перший дрон стає на значення 2, другий на 4. Відповідно одразу вони не стикаються, тому програма переміщує їх вперед чи назад по значеннях масиву. Так як перший дрон був на 2, то він переміщується на 2 вперед, а другий на 4 назад. Отже на визоді маємо значення 3 та 0, тому виводиться Miss.

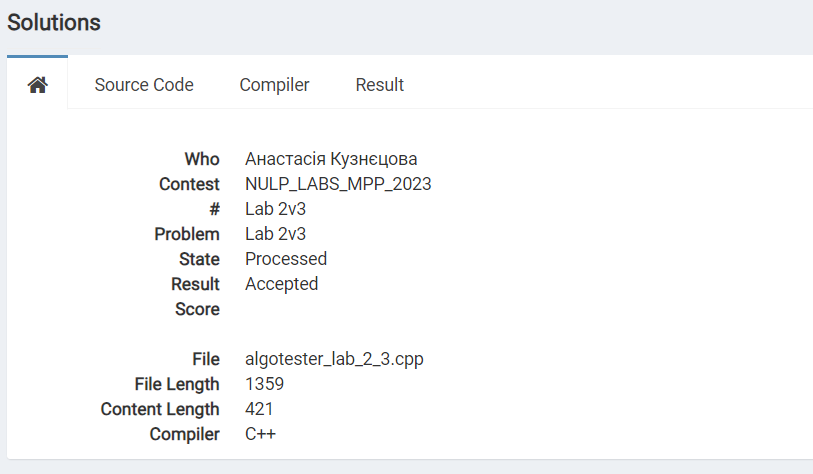
Час, затрачений на виконання завдання: 2 дні

Рисунок 6 Скріншот Algotester 2

Завдання №4 **Algotester Lab 3**

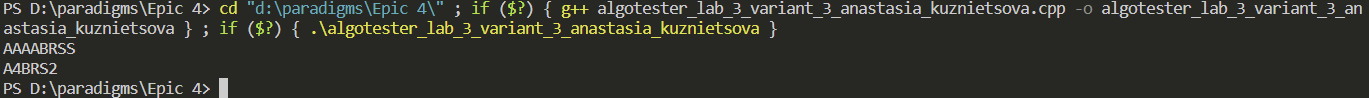


Рисунок 7 Виконання Algotester Lab 3

Програма зчитує введені значення. Так як А повторилось 4 рази, вона виводить А4. Так само з наступними значеннями.

Час, затрачений на виконання завдання: 6 годин

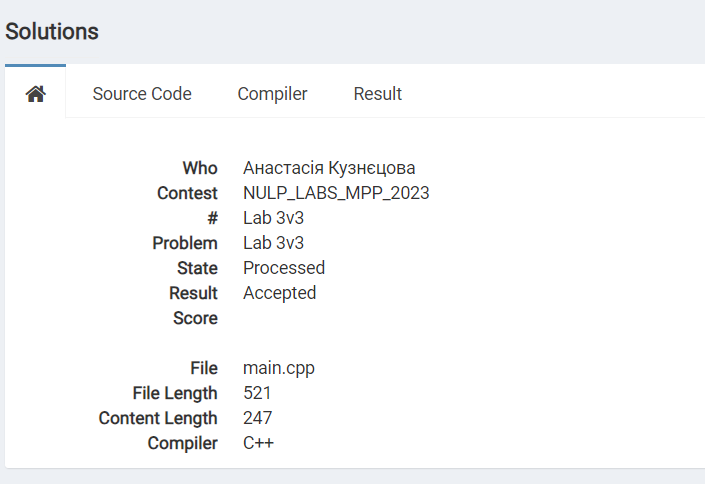


Рисунок 8 Скриншот Algotester 3

Завдання №5 **Class Practice Work**

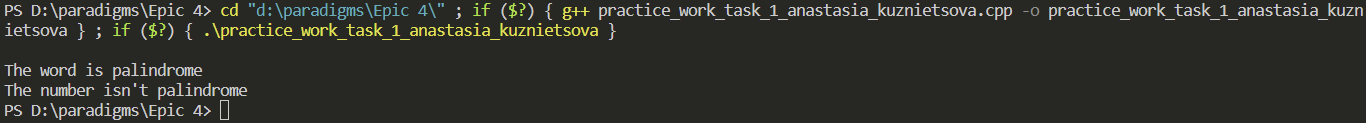


Рисунок 9 Виконання Class Practice Work

У функції main введене слово “level” – що є паліндромом. Пройшовшись по всіх літерах слова, програма видала що воно є паліндромом. А число 263962 – не є паліндромом. Так як це тип int програма перекидає користувача у функцію, що викликається цим типом, і перетворює тип int на srting, після чого викликає функцію для визначення чи є стрічна паліндромом.

Час, затрачений на виконання завдання: 7 годин

# **Pull-request:** https://github.com/artificial-intelligence-department/ai\_programming\_playground/pull/623

# **Висновки:**

Опрацьовано оптимізацію обробки масивів та вивчено прості структури даних. Ознайомлено з одновимірними та двовимірними масивами, на основі яких виконувались лабораторні роботи.