

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



Звіт

про виконання

Лабораторних та практичних робіт № 7

з дисципліни: «Мови та парадигми програмування»

з розділу: «Розробка, програмування та код. Середовища для розробки.»

Виконав:

студент групи ШІ-14

Павлюк Максим Сергійович

Тема роботи:

Розв'язання різних задач.

Мета роботи:

Ознайомитися з основними математичними задачами, які можуть бути розв'язані за допомогою програмування.

Виконання роботи:

1. Опрацювання завдання та вимог до програм та середовища:

Завдання №1 VNS PW 1

- 11

Варіант 11. $p = \frac{a}{b}$, де $a = (1-y) \frac{(x+y)^2 / (x+4)^3}{e^{-(x-2)} + (x^3+4)}$;

$$b = \frac{1 + \cos(y-2)}{x^{4/2} + \sin^2(y-2)}; x=1,25; y=0,93.$$

- Важливі деталі для врахування в імплементації програми - ніяких які би можна було виділити

Завдання №2 VNS PW 2

- 24

Варіант 24. Підрахувати, скільки разів функція $y = \cos x^2 \cdot e^{-x}$

приймає негативне значення, якщо $x \in [0,3;5]$; $h_x = 0,1$.

- Важливі деталі для врахування в імплементації програми – ніяких які би можна було виділити

Завдання №VNS PW 3

- 5

Варіант 5. Обчислення вартості покупки складатиметься з

11

декількох зошитів і такої ж кількості обкладинок до них. Нижче приведений вид екрану під час виконання програми, що рекомендується (дані, які вводяться користувачем, виділені напівжирним шрифтом).

Обчислення вартості покупки.

Введіть початкові дані:

Ціна зошита (грн.) > **2.75**

Ціна обкладинок (грн.) > **0.5**

Кількість комплектів (грн.) > **7**

Вартість покупки: 15.45 грн.

Обчислення об'єму і площі поверхні кулі, радіусом r , за

формулами $V = \frac{3}{4} \cdot \pi \cdot r^3$; $S = 4 \cdot \pi \cdot r^2$.

- Важливі деталі для врахування в імплементації програми – ніяких які би можна було виділити

Завдання №4 VNS PW 4

- 14

- Деталі завдання: Скласти програму, яка вводиться з клавіатури послідовність з п'яти дробових чисел і після введення кожного числа виводить середнє арифметичне введеній частині послідовності.
- Важливі деталі для врахування в імплементації програми – ніяких які би можна було виділити

2. Дизайн та планована оцінка часу виконання завдань:

Програма №1 VNS PW 1



Рисунок 1 блок-схема до завдання 1

- Планований час на реалізацію – 10 хв
- Важливі деталі для врахування в імплементації програми – ніяких які би можна було виділити

Програма №2 VNS PW 2

- Блок-схема

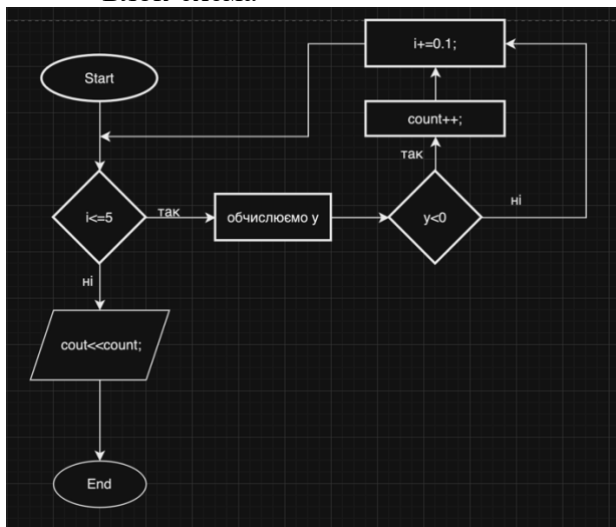


Рисунок 2 блок-схема до завдання 2

- Планований час на реалізацію – 20 хв
- Важливі деталі для врахування в імплементації програми – ніяких які би можна було виділити

Програма №3 VNS PW 3



Рисунок 3 блок-схема до завдання 3

- Планований час на реалізацію – 30 хв
- Важливі деталі для врахування в імплементації програми – ніяких які би можна було виділити

Програма №4 VNS PW 4

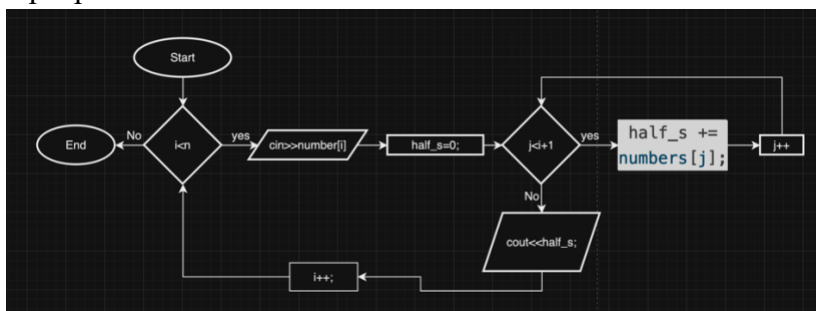


Рисунок 4 блок-схема до завдання 4

- Планований час на реалізацію – 30 хв
- Важливі деталі для врахування в імплементації програми – ніяких які би можна було виділити

3. Код програм з посиланням на зовнішні ресурси:

Завдання №1 VNS PW 1

```

1  #include<iostream>
2  #include<cmath>
3  using namespace std;
4
5  int main() {
6      // Ініціалізація змінних
7      float x = 1.25, y = 0.93;
8      float a = 0, b = 0, p, temp1, temp2;
9
10     // Обчислення значення a
11     temp1 = pow((x + y), 2) / pow((x + 4), 3);
12     temp2 = -1 * pow(M_E, (-1 * x + 2)) + pow(x, 3) + 4;
13     a = (1 - y) * temp1 / temp2;
14
15     // Виведення значення a
16     cout << "a = " << a << endl;
17
18     // Обчислення значення b
19     temp1 = 1 + cos(y - 2);
20     temp2 = pow(x, 2) + pow(sin(y - 2), 2);
21     b = temp1 / temp2;
22
23     // Виведення значення b
24     cout << "b = " << b << endl;
25
26     // Обчислення та виведення значення p
27     p = a / b;
28     cout << "p = " << p << endl;
29
30     return 0;
31 }
32

```

Завдання №2 VNS PW 2

```

1  #include<iostream>
2  #include<cmath>
3  using namespace std;
4
5  int main() {
6      // Ініціалізація змінних
7      double x = 0.3, y;
8      int count = 0;
9
10     // Цикл для обчислення функції та підрахунку від'ємних значень
11     for (double i = 0.3; i <= 5; i = i + 0.1) {
12         // Обчислення значення функції y = cos(i^2) * e^(-i)
13         y = cos(pow(i, 2)) * pow(M_E, -1 * i);
14
15         // Перевірка, чи отримане значення менше нуля
16         if (y < 0) {
17             count++;
18         }
19     }
20
21     // Виведення кількості від'ємних значень
22     cout << count << endl;
23
24     return 0;
25 }
26

```

Завдання №3 VNS PW 3

```
1 #include<iostream>
2 #include<cmath>
3 using namespace std;
4
5 int main() {
6     // Частина 1 завдання 5
7     float price_nb = 0, price_cov = 0;
8     int amount_nb = 0;
9
10    // Введення початкових даних
11    cout << "Введіть початкові дані: " << endl;
12    cout << "Ціна зошита (грн.) > ";
13    cin >> price_nb;
14    cout << "Ціна обкладинки (грн.) > ";
15    cin >> price_cov; (const char [43])"Кількість комплектів > "
16    cout << "Кількість комплектів > ";
17    cin >> amount_nb;
18
19    // виведення вартості покупки
20    cout << "Вартість покупки: " << amount_nb * (price_cov + price_nb) << " грн. " << endl;
21}
```

```
// Частина 2
float r, S, V;

// Введення значення r
cout << "Введіть r: ";
cin >> r;

// Обчислення та виведення S
S = 4 * M_PI * r * r;
cout << "S = " << S << endl;

// Обчислення та виведення V
V = 0.75 * M_PI * pow(r, 3);
cout << "V = " << V << endl;

return 0;
}
```

Завдання №4 VNS PW 4

```
1 #include<iostream>
2 // #include<cmath>
3 using namespace std;
4
5 int main() {
6     // Ініціалізація змінних
7     int n = 5;
8     double half_s = 0;
9     double numbers[n];
10
11    // Зчитування та обчислення значень для кожного введенного числа
12    for (int i = 0; i < n; i++) {
13        cin >> numbers[i];
14        half_s = 0;
15
16        // Обчислення середнього арифм. суми чисел
17        for (int j = 0; j < i + 1; j++) {
18            half_s += numbers[j];
19        }
20        cout << half_s / 2 << endl;
21    }
22
23    return 0;
24 }
```

Pull Request - https://github.com/artificial-intelligence-department/ai_programming_playground/pull/848

4. Результати виконання завдань, тестування та фактично затрачений час:

Завдання №1

```
a = 0.000599295
b = 0.634707
p = 0.000944207
max@Damashchuk-MBP
```

Рисунок 5 виконання програми 1

Час затрачений на виконання завдання – 10 хв

Завдання №2

```
Negative function values: 21
max@Damashchuk-MBP epic 7 %
```

Рисунок 6 виконання програми 2

Час затрачений на виконання завдання – 30 хв

Завдання №3

```
ns_practice_task_3_максум_рав
Введіть початкові дані:
Ціна зошита (грн.) > 2.73
Ціна обкладинок (грн.) > 0.11
Кількість комплектів > 100
Вартість покупки: 284 грн.
Введіть r: 5
S = 314.159
V = 294.524
max@Damashchuk-MBP epic 7 %
```

Рисунок 7 виконання програми 3

Час затрачений на виконання завдання – 50 хв

Завдання №4

```
ns_practice_task_4_максум_
5
Half right now is: 2.5
5
Half right now is: 5
5
Half right now is: 7.5
5
Half right now is: 10
5
Half right now is: 12.5
max@Damashchuk-MBP epic 7 %
```

Рисунок 8 виконання програми 4

Час затрачений на виконання завдання – 30 хв

5. Контрольні запитання:

1. Назвіть основні властивості алгоритму.

Дискретність, визначеність, виконуваність, скінченність, результативність, масовість, ефективність.

2. Що таке алгоритм?

Це визначена послідовність кроків, для вирішення певної задачі.

3. Визначте основні етапи розробки алгоритмів.

1) аналіз задачі

2) вибір методу рішення

3) розробка алгоритму

4. Перелічить базові конструкції.

Розгалуження, цикли, функції.

5. Складні базові конструкції

Вкладені цикли, розгалуження; перевантажені функції.

6. Дайте визначення конструкції розгалуження.

Конструкції які виконують код в них, тільки за певних умов.

7. Дайте визначення конструкції цикл.

Цикл – виконання частини коду певну кількість разів доки не задовільниться умова.

8. Сформулюйте правило виконання циклу з передумовою.

Цикл буде виконуватись – доки певна умова є істинною.

9. Сформулюйте правило виконання циклу з відомою кількістю повторювань тіла циклу.

Цикл буде виконуватись певну кількість разів.

10. Що таке обчислювальна складність алгоритму?

Кількість ресурсів необхідна для виконання алгоритму.

11. Як оцінити обчислювальну складність?

Проводять оцінку необхідних для алгоритму часу та пам'яті при збільшенні вхідних даних.

12. Рекурсивні функції. Переваги їх використання.

Функція – яка викликає саму себе. Перевага у розробці завдань які можуть бути розбиті на менші завдання. Наприклад завдання на сортування, потребують використання рекурсії.

Висновки:

Навчився вирішувати базові математичні задачі за допомогою програмування. Повторив основні алгоритмами при вирішуванні задач.