

МІНІСТЕРСТВО ОСВІТИ І НАУКИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА
ПОЛІТЕХНІКА»

Кафедра систем штучного інтелекту



Лабораторна робота №1
З курсу “Алгоритмізація та програмування”

Виконала:
ст.гр. КН-110
Ямнюк Аліна

Львів – 2018

Варіант 7

1. Постановка завдання:

Обчислити значення виразу при різних дійсних типах даних (float й double). Обчислення варто виконувати з використанням проміжних змінних. Порівняти й пояснити результати.

Обчислити значення виразів. Пояснити отримані результати.

2. Програма розв'язку завдання 1:

1. #include <stdio.h>

2. #include <math.h>

3.

4. int main(){

5. float a=1000;

6. float b=0.0001;

7. double a1=1000;

8. double b1=0.0001;

9. float result1=(pow(a-b,3)-pow(a,3))/(pow(b,3)-3*a*pow(b,2)-3*pow(a,2)*b);

10. double result2=(pow(a1-b1,3)-pow(a1,3))/(pow(b1,3)-3*a*pow(b1,2)-3*pow(a1,2)*b1);

11. printf("%f\n", result1);

12. printf("%f\n", result2);

13.

14. }

3. Результати роботи програми для даних типу float та для даних типу double :

```
jharvard@appliance (~): make lab1-1
clang -ggdb3 -O0 -std=c99 -Wall -Werror lab1-1.c -lcs50 -lm -o lab1-1
jharvard@appliance (~): ./lab1-1
1.220703
1.000000
jharvard@appliance (~): █
```

4. Пояснення результатів:

Float: 1.220703

Double: 1.000000

Результати вийшли різні, бо типи Float і Double мають різну точність. При обчисленнях з типом Float точність менша, і при кожному обчисленні похибка накопичується. Тип даних Double більш точний, ніж Float.

5. Програма розв'язку завдання 2:

1.#include <stdio.h>

2.

3.int main()

4.{

5.int m,n;

6.float result1, result2, result3;

7.printf("Input m: ");

8 scanf("%d", &m);

9.printf("Input n: ");

```
10 scanf("%d", &n);
11.
12 result1 = m + --n;
13 result2 = m++ < ++n;
14 result3 = n-- < --m;
15.
16 printf("%f\n", result1);
17 printf("%f\n", result2);
18 printf("%f\n", result3);
19 }
```

6. Результати роботи програми 2:

```
jharvard@appliance (~): make lab1-2
clang -ggdb3 -O0 -std=c99 -Wall -Werror lab1-2.c -lcs50 -lm -o lab1-2
jharvard@appliance (~): ./lab1-2
Input m: 2
Input n: 3
4.000000
1.000000
0.000000
jharvard@appliance (~): █
```

7. Пояснення результатів:

Нехай $m=2$, $n=3$. При виконанні першої дії спочатку змінна n зменшується на одиницю, потім додається до m , в результаті отримуємо 4. При виконанні другої дії спочатку до значення $n=2$ додається одиниця, тоді порівнюється зі

значенням m , в результаті отримуємо, що $m < n$, тобто справді $2 < 3$. Це значення є `True`, тому компілятор показав 1.

При виконанні третьої дії від значення $m=3$ віднімається одиниця, тоді порівнюється зі значенням n , в результаті отримуємо, що $n < m$. Це значення є `False`, адже 3 не є менше за 2, тому компілятор показав 0.

Прогрес на курсі CS50:

```
1#include <cs50.h>
2#include <stdio.h>
3
4int main()
5{
6    int height
7    do
8    {
9        printf("Print height of the pyramid no more than 23: ");
10       height = GetInt();
11    }
12    while ((height > 23) || (height <= 0));
13
14    for (int i=0; i<height; i++)
15    {
16        int space, hash;
```

Натисніть ESC та TAB або клацніть за межами редактора коду, щоб вийти

Обробка