

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

Кафедра систем штучного інтелекту

**Лабораторна робота
№1**

з дисципліни
«Алгоритмізації та програмування»

Виконав:
студент групи КН-108
Жеребецький Олег

Львів – 2018 р.

5. Зміст звіту

1. Постановка завдання.
2. Програма розв'язку завдання1.
3. Результати роботи програми для даних типу float.
4. Результати роботи програми для даних типу double.
5. Пояснення результатів.
6. Програма розв'язку завдання2.
7. Результати роботи програми.
8. Пояснення результатів.

Постановка завдання

- Обчислити значення виразу
$$\frac{(a+b)^2 - (a^2 + 2ab)}{b^2}$$
 при різних дійсних типах даних (float й double).
- Порівняти й пояснити отримані результати.
- Обчислити значення виразів: --m-++n ; m*n<n++ ; n-- > m++ ;
- Пояснити отримані результати.

Програма розв'язання завдання №1

```
#include <stdio.h>
#include <math.h>

int main (void)
{
    float a1, b1;
    double a2, b2;

    printf("Give me two numbers for floats\n");
    scanf("%f %f",&a1, &b1);
    printf("Float out :%f\n",(pow(a1+b1,2) - (pow(a1, 2)+ 2*a1*b1))/pow(b1,2));

    printf("Give me two numbers for double\n");
    scanf("%lf %lf",&a2, &b2);
    printf("Double out :%lf\n",(pow(a2+b2,2) - (pow(a2, 2)+ 2*a2*b2))/pow(b2,2));
```

```
    return 0;  
}
```

Результат роботи для даних типу Float

```
jharvard@appliance (~/.Dropbox/Labs): ./lab1-1  
Give me two numbers for floats  
1000 0.001  
Float out : -46874.041873
```

Результат роботи для даних типу Double

```
Give me two numbers for double  
1000 0.001  
Double out : 1.0000008
```

Пояснення результатів

При виконанні дій такого масштабу як множення або підняття до степеня Числа які мають соті, тисячні і тд. , виходять за межі визначеності типу Float швидше ніж типу Double, в наслідок чого відбувається різне заокруглення, а внаслідок цього різні числа, що різко міняє відповідь у великих виразах.

Програма розв'язання завдання №2

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    int n, m, m1, n1;
```

```
    /* Inputing*/
```

```
    printf("Give me two numbers\n");
```

```
    scanf("%i %i",&n, &m);
```

```
    /* First task*/
```

```
    printf("Answers:\n");
```

```

n1=n;m1=m;
printf("%i\n",--m1-++n1 );
/* Second task*/
n1=n; m1=m;
printf("%s\n", (m1*n)<n1++?"True":"False");
/* Third task*/
n1=n; m1=m;
printf("%s\n", n1-->m1++?"True":"False");

return 0;
}

```

Результати роботи програми

```

Give me two numbers
10
-25
Answers:
-37
True
True

```

Пояснення результатів

У 2 та 3 рядку відбувається зчитування двох чисел n та m відповідно , котрі будуть використовуватись при розв'язанні формул в подальшому.

У п'ятому рядку виводиться значення формули $--m-++n$, оскільки дії $--$ та $++$ стоять перед змінними m та n відповідно, ця формула буде обрахована уже зі зміненими m та n (тобто $(-25-1)-(10+1) = -37$).

У шостому рядку виводиться значення нерівності $m*n < n++$, а саме правдива вона , чи ні. Оскільки у даній нерівності дія $++$ стоїть після відповідного n , то це n не буде збільшене у під час обрахунку цієї нерівності.

У сьомому рядку виконується аналогічна операція тільки для формули $n-- > m++$.

P.S. У цій програмі для кожної з трьох дій присвоювались ті ж значення n та m , що були введені на початку. Можна було використовувати ті ж , у зв'язку з цим у другий дії використовувались уже $--m$ тобто $m - 1$ та $n + 1$ відповідно, адже над ними були проведенні відповідні операції у попередньому прикладі. Але це не було описано в умові завдання).