

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТУ “ЛЬВІВСЬКА
ПОЛІТЕХНІКА”

Кафедра систем штучного інтелекту

Лабораторна робота №1

з дисципліни

«Алгоритмізація та програмування»

Виконав:

студент групи КН-109

Яворський Володимир

Викладач:

Гасько Р.Т

Львів – 2018 р.

Лабораторна робота №1.

Тема: "Знайомство з С. Виконання програми простої структури"

Мета: Знайомство з середовищем програмування, створення, відлагодження й виконання простої програми, що містить ввід/вивід інформації й найпростіші обчислення.

Варіант №11

Постановка завдання №1:

Обчислити значення виразу при різних дійсних типах даних (float й double).

Обчислення варто виконувати з використанням проміжних змінних.

Порівняти й пояснити отримані результати.

$$\frac{(a-b)^4 - (a^4 - 4a^3b)}{6a^2b^2 - 4ab^3 + b^4}$$

при a=100, b=0.001

Код програми :

- З використанням ***float***:

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

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

```
int main ()
```

```
{
```

```
    float a, b, c;
```

```
    printf ("Input a:");
```

```
    scanf ("%f", &a);
```

```
    printf ("Input b:");
```

```
    scanf ("%f", &b);
```

```
    c = ((a - b)*(a - b)*(a - b)*(a - b)-(pow(a, 4) - 4*pow(a, 3)*b))
```

```
/ (6*pow(a, 2)*pow(b, 2) - 4*a*pow(b, 3) + pow(b, 4));
```

```
printf ("Result: %f\n", c);
```

```
return 0;
```

```
}
```

- З використанням **double**:

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

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

```
int main ()
```

```
{
```

```
double a, b, c;
```

```
printf ("Input a:");
```

```
scanf ("%lf", &a);
```

```
printf ("Input b:");
```

```
scanf ("%lf", &b);
```

```
c = ((a - b)*(a - b)*(a - b)*(a - b)-(pow(a, 4) - 4*pow(a, 3)*b))
```

```
/ (6*pow(a, 2)*pow(b, 2) - 4*a*pow(b, 3) + pow(b, 4));
```

```
printf ("Result: %lf\n", c);
```

```
return 0;
```

```
}
```

Результат виконання програми:

```
jharvard@appliance (~/programs): ./lab1
Input a:100
Input b:0.001
Result: 0.003167
jharvard@appliance (~/programs):
```

Float

```
./jharvard@appliance (~/programs): ./lab1
Input a:100
Input b:0.001
Result: 1.000000
jharvard@appliance (~/programs):
```

Double

Пояснення результатів:

Оскільки тип **double** містить у два рази більше знаків після коми, ніж **float**, то обрахунки **double** є більш точними, але цей тип займає у оперативній пам'яті більше байтів.

Постановка завдання №2:

Обчислити значення виразів. Пояснити дані результати.

1) $n++ * m$

2) $n++ < m$

3) $m-- > m$

Код програми :

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

```
int main()
```

```
{
```

```
    int a, b, c, m, n;
```

```
    printf("input n:");
```

```
    scanf("%d", &n);
```

```
    printf("input m:");
```

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

```
    a = n++ * m;
```

```
    printf("%d++ * %d = %d \n", n, m, a);
```

```
    printf("input n:");
```

```
    scanf("%d", &n);
```

```
    printf("input m:");
```

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

```
    b = n++ < m;
```

```

printf("%d++ < %d = %d \n", n, m, b);

printf("input n:");
scanf("%d", &n);
printf("input m:");
scanf("%d", &m);

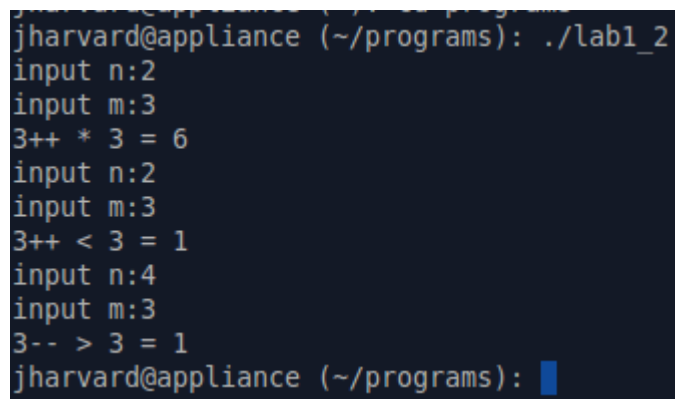
c = n-- > m;

printf("%d-- > %d = %d \n", n, m, c);

return 0;
}

```

Результат виконання програми:



```

jharvard@appliance (~/programs): ./lab1_2
input n:2
input m:3
3++ * 3 = 6
input n:2
input m:3
3++ < 3 = 1
input n:4
input m:3
3-- > 3 = 1
jharvard@appliance (~/programs):

```

Пояснення результатів:

- 1) Оскільки інкремент ++ стоїть після числа, то під час обраховування виразу береться його початкове значення (2), а лише після обчислення **n** отримує значення 3. Тому **2 * 3 = 6**.
- 2) Оскільки інкремент ++ стоїть після числа, то під час обраховування виразу береться його початкове значення (2), а лише після обчислення **n** отримує значення 3. Тому **2 < 3** є вірним твердженням (1).
- 3) Оскільки інкремент ++ стоїть після числа, то під час обраховування виразу береться його початкове значення (4), а лише після обчислення **n** отримує значення 3. Тому **4 > 3** є вірним твердженням (1).

