Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное учреждение высшего образования  
«Пермский национальный исследовательский политехнический университет»

Электротехнический факультет  
Кафедра «Информационные технологии и автоматизированные системы»

**О Т Ч Ё Т**

Основы алгоритмизации и программирования

Семестр 1

Лабораторная работа №1

Выполнил  
студент группы РИС-21-2б  
Елькин Андрей Владиславович

Проверил  
доцент кафедры ИТАС  
Полякова О.А.

Пермь, 2021

## 1 Постановка задачи

## Вычислить значение выражения при различных вещественных типах данных (float и double). Вычисления следует выполнять с использованием промежуточных переменных. Сравнить и объяснить полученные результаты.

,

при а=100, b=0.001

1. Вычислить значения выражений. Объяснить полученные результаты.

n++ \*m

n++ < m

m-- > m

## 2 Программа решения задания 1

#include <iostream>

using namespace std;

int main() {

setlocale(LC\_ALL, "ru");

float af = 100.0, bf = 0.001;

float answerf1 = (af - bf) \* (af - bf) \* (af - bf) \* (af - bf);

float answerf2 = af \* af \* af \* af - 4 \* af \* af \* af \* bf;

float answerf3 = answerf1 - answerf2;

float answerf4 = 6 \* af \* af \* bf \* bf;

float answerf5 = 4 \* af \* bf \* bf \* bf;

float answerf6 = bf \* bf \* bf \* bf;

float answerf7 = answerf4 - answerf5 + answerf6;

float answerf = answerf3 / answerf7;

cout << answerf1 << endl << answerf2 << endl << answerf3 << endl << answerf4 << endl << answerf5 << endl << answerf6 << endl << answerf7 << endl << answerf << endl << endl;

double ad = 100.0, bd = 0.001;

double answerd1 = (ad - bd) \* (ad - bd) \* (ad - bd) \* (ad - bd);

double answerd2 = ad \* ad \* ad \* ad - 4 \* ad \* ad \* ad \* bd;

double answerd3 = answerd1 - answerd2;

double answerd4 = 6 \* ad \* ad \* bd \* bd;

double answerd5 = 4 \* ad \* bd \* bd \* bd;

double answerd6 = bd \* bd \* bd \* bd;

double answerd7 = answerd4 - answerd5 + answerd6;

double answerd = answerd3 / answerd7;

cout << answerd1 << endl << answerd2 << endl << answerd3 << endl << answerd4 << endl << answerd5 << endl << answerd6 << endl << answerd7 << endl << answerd << endl << endl;;

system("Pause");

return 0;

**}**

## 3 Результаты работы программы для данных типа float

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| 99996000.0 | 99996000.0 | 0.00000000 | 0.0599996075 | 0.00000000 |

# 4 Результаты работы программы для данных типа double

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| 99996000.059999585 | 99996000.000000000 | 0.059999585151672363 | 0.059999600000999999 | 0.99999975250955619 |

# 

# 5 Объяснение результатов

Результаты получились кардинально разными. Всё началось с первого выражения , в котором должно получиться 99996000, но в программе с double получилось число с цифрами после мантиссы. Это вышло из-за того, что компьютер работает не с десятичными числами, а с двоичными. При выполнении вычитания в первом действии компьютер перевел числа a и b в двоичную систему счисления, после чего вычел из числа a число b. Получившееся число из двоичной системы счисления было переведено в десятичную. Из-за этих действий при вычитании у числа появилась небольшая погрешность. У программы, в которой использовался тип float этой погрешности не возникло, потому что float вмешает в себя числа от +/- 3.4E-38 до 3.4E+38, а получившаяся в результате вычитания погрешность выходит за эти границы. В итоге погрешность не записалась, потому что на неё не хватило памяти.

В случае с типом double погрешность при вычитании была записана, потому что double включает числа в диапазоне от +/- 1.7E-308 до 1.7E+308. А далее при возведении в степень погрешность значительно выросла, что и привело к расхождению результатов в 0.059999585

Далее из выражения , вычитают выражение, в котором нет погрешности и в обоих случаях получается 99996000.

Так в выражении с типом float из 99996000 вычитают 99996000, что в результате дает 0, а в выражении с типом double из 99996000.059999585 вычитают 99996000, что в результате дает 0.059999585. Далее в обоих случаях получившиеся числа делятся на выражение , в котором из-за сложения и вычитания также возникает погрешность, но в отличии от первой ситуации выражение не возводится в степень, поэтому погрешность очень мала.

В конечном итоге выражение с типом float равно 0, так как 0 делится на 0.0599996075, а выражение с типом double практически равно единице, так как 0.059999585 делится на 0.059999600000999999.

# 6 Программа решения задания 2

#include <iostream>

using namespace std;

int main() {

setlocale(LC\_ALL, "ru");

int n, m, ans1;

bool ans2, ans3;

cout << "Введите значение числа n : ";

cin >> n;

cout << "Введите значение числа m : ";

cin >> m;

ans1 = n++ \* m;

ans2 = n++ < m;

ans3 = m-- > m;

cout << "Значение выражения n++ \* m : " << ans1 << endl;

cout << "Значение выражения n++ < m : " << ans2 << endl;

cout << "Значение выражения m-- > m : " << ans3 << endl;

system("Pause");

return 0;

}

# 7 Результаты работы программы

Введите значение числа n : 2

Введите значение числа m : 2

Значение выражения n++ \* m : 4

Значение выражения n++ < m : 0

Значение выражения m-- > m : 0

# 8 Объяснение результатов

Выражение, содержащее операции сравнения возвращают 1, если выражение истинно, и 0, если выражение ложно.

Первое выражение равно 4, так как в начале идет переменная n, равная 2. Далее к n при помощи постфиксного инкремента прибавляется 1, но так как он постфиксный, то 1 прибавится только после того, как выражение закончится. Далее следует знак умножения и переменная m, равная 2. Получается, что выражение равно 2 умножить на 2, то есть 4, а переменная n станет равна 3.

Второе выражение ложно, поэтому вывелся 0. В начале идет переменная n, равная 3. Далее к n при помощи постфиксного инкремента прибавляется 1, но так как он постфиксный, то 1 прибавится только после того, как выражение закончится. Далее при помощи знака меньше n сравнивается с m. Приложение выводит 0, так как 3 больше 2, а в выражении сказано, что 3 меньше 2. В итоге выведется 0, а переменная n равна 4.

Третье выражение ложно, поэтому вывелся 0. В начале идет переменная m, равная 1. Далее от m при помощи постфиксного декремента вычитается 1, но так как он постфиксный, то 1 прибавится только после того, как выражение закончится. Далее при помощи знака больше m сравнивается с m. Приложение выводит 0, так как m равно m, а в выражении сказано, что m больше m. В итоге выведется 0, а переменная m равна 0.