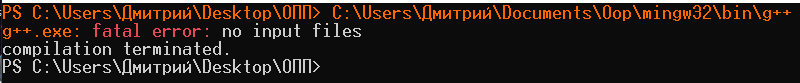
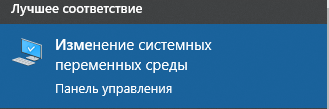
ООП

Задание 1

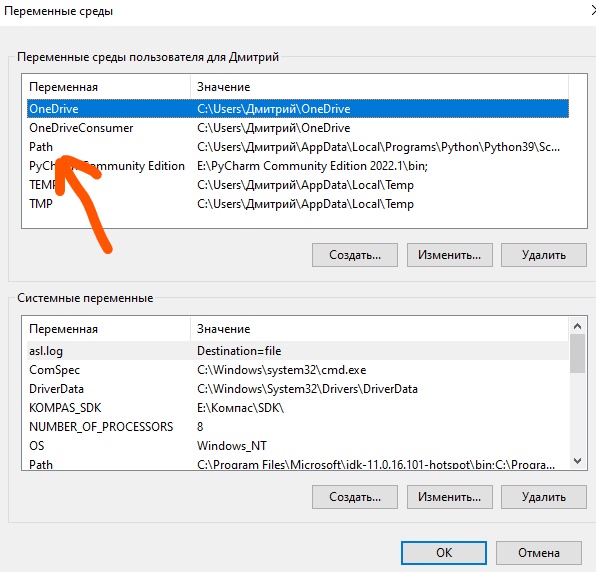
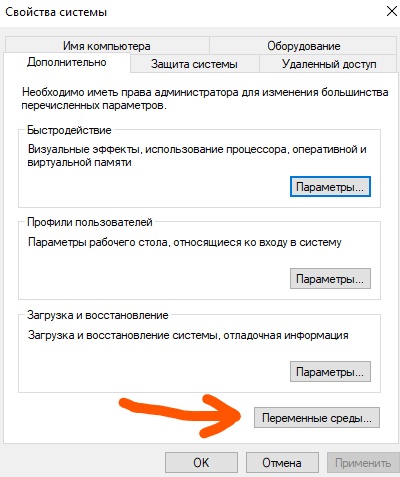
После того как вы скачали компилятор на свой пк, чтобы его открыть вам нужно будет прописывать полностью его путь в консоли  (PowerShell)



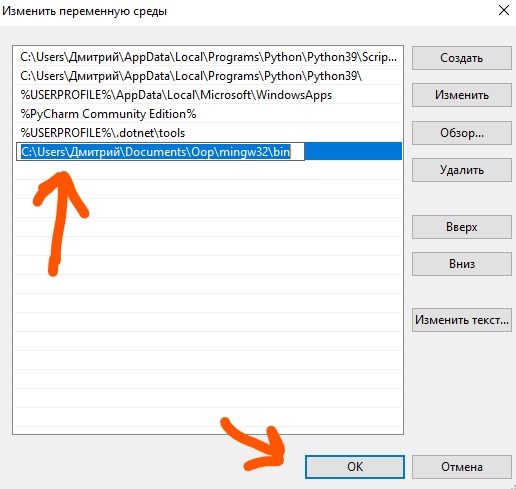
Для облегчения этого процесса мы можем добавить путь к компилятору в переменную среды окружения Path. Для этого в поиске найдём “Изменение системных переменных среды” ( Или “Изменение переменных среды текущего пользователя”, если работаете не на своём компьютере)



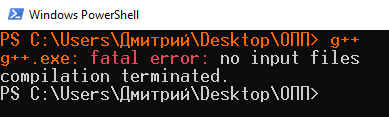
После открытия переходим в раздел “Переменный среды”, и открываем двойным кликом переменную Path.



В появившемся окне мы вставляем ссылку до нашего компилятора в свободную ячейку, после нажимаем “Ок” и выходим с данного окна.



Всё готово теперь для обращения к компилятору нам не надо будет каждый раз прописывать путь до него, можем просто написать его название



Если у нас уже есть написанная программа, нам нужно её скомпилировать, для этого в консоли нужно обратиться к самому компилятору и через пробел указать путь до файла с вашей программой ( или просто название файла, если вы открыли компилятор уже с папки, где находится ваша программа).

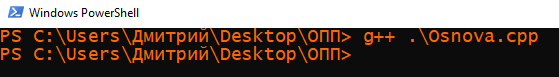


Или если уже уставлен путь до папки с нашей программой



Пример:”=

Допустим у нас есть написанная программа Osnova.cpp, для её компилирования мы напишем в консоли “g++ .\Osnova.cpp” (.\ здесь нужна, так как мы обращаемся к пути файла)

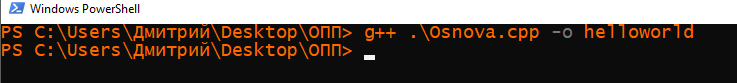


После запуска наша программа скомпилируется и появится файл с расширением .exe

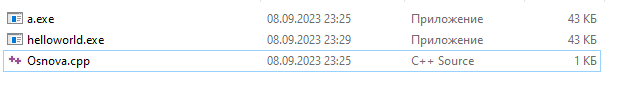


По умолчанию это файл с названием a.exe, но мы также можем изменить данное название, если при компиляции после название файла с программой напишем:

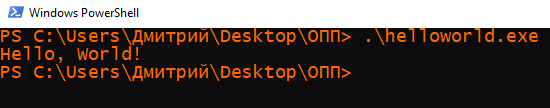
..“название\_ файла\_кода\_ программы” –o “название\_скомпилированной\_программы”



После этого у нас появится файл helloworld.exe, как и указали



Файлы a.exe и helloworld.exe это уже откомпилированные программы, которые мы можем запускать, если в консоли просто напишем названием файла ( с указанием «.\», если мы находимся в папке с файлом, иначе указываем весть путь)



**Задание 2.**

Задача **27**Даны действительные положительные числа a, b, c. По трем сторонам с длинами a, b, c можно построить треугольник. Найти углы треугольника.

Код программы

/\*27 Даны действительные положительные числа a, b, c.

По трем сторонам с длинами a, b, c можно построить

треугольник. Найти углы треугольника.\*/

#include <iostream>

#include <cmath>

using namespace std;

// Проверяет образуют ли введённые стороны теугогльник

// Должно выполняться условие a + b > c

int triagle\_check(float a, float b, float c)

{

if ((a + b <= c) || (a + c <= b) || (c + b <= a)) return 0;

else return 1;

}

// Расчет угла по его сторонам через теорему косинусов

float search\_angle(float a, float b, float c)

{

return (acos((a\*a + b\*b - c\*c)/(2\*a\*b)))\*180/M\_PI;

}

int main()

{

// Переменные a,b,c для хранения длин сторон

float a,b,c;

// Переменные ang1,ang2,ang3 для хранения углов

float ang1,ang2,ang3;

// Не смог разобраться как включить русский язык

// Пробовал и SetLocale и SetConsoleCP(1251)

// И шрифты менял в консоли, не помогло(

// Вводим длины сторон

cout << "Side a = ";

cin >> a;

cout << "Side b = ";

cin >> b;

cout << "Side c = ";

cin >> c;

// Проверяем образуют ли стороны треугольник

if (triagle\_check(a,b,c) != 1) cout << "This is not Triagle";

else

{

// Ищем углы по формуле

ang1 = search\_angle(a,b,c);

ang2 = search\_angle(a,c,b);

ang3 = 180 - ang1 - ang2;

cout << "Angle 1 = " << ang1 << endl;

cout << "Angle 2 = " << ang2 << endl;

cout << "Angle 3 = " << ang3 << endl;

}

}

Проверка

