# Код

/\*

Координаты и радиус предполагаются цеоыми. A - центр одного из оснований конуса, B - другого, R - радиуса основания.

Точка X лежит внутри цилиндра, если выполняются два условия:

1. точка X лежит между плоскостями оснований

2. расстояние от X до AB меньше R

Первое равносильно тому, что два угла XAB и XBA не меньшн 0 и не больше ph/2, или, что тоже самое, скалярное произведение сторон угла (как векторов) больше 0.

Второе - рассматриваем треугольник AXB, его площадь - это полсумма основания на высоту, либо полсумма сторон на синус угла. Отсюда можно выразить высоту, которая и есть расстояние от X до AB.

\*/

#include <iostream>

using namespace std;

struct point {

int x;

int y;

int z;

};

struct cylinder {

point a;

point b;

int radius;

};

void inputPoint(point &A) {

cout << "Enter point coordinates. x, y, z:\n";

cin >> A.x >> A.y >> A.z;

}

void printPoint(point A) {

cout << "Point coordinates: " << A.x << " " << A.y << " " << A.z << endl;

}

void inputCylinder(cylinder& c) {

cout << "Enter coordinates of A and B:\n";

inputPoint(c.a);

inputPoint(c.b);

cout << "Enter radius of the base\n";

cin >> c.radius;

}

void printCylinder(cylinder C) {

cout << "";

}

int dotProduct(point a, point b, point c, point d) {

//calculates dot product of vectors ab and cd

return (b.x - a.x) \* (d.x - c.x) + (b.y - a.y) \* (d.y - c.y) + (b.z - a.z) \* (d.z - c.z);

}

bool isInCylinder(point x, cylinder c) {

if ((dotProduct(c.a, x, c.a, c.b) < 0) || (dotProduct(c.b, x, c.b, c.a) < 0)) {

return false;

}

if ((dotProduct(c.a, x, c.a, x) \* dotProduct(c.a, c.b, c.a, c.b) - dotProduct(c.a, x, c.a, c.b) \* dotProduct(c.a, x, c.a, c.b)) > c.radius \* c.radius \* dotProduct(c.a, c.b, c.a, c.b)) {

return false;

}

return true;

}

int main()

{

point x = { 0,0,0 };

cylinder c;

cout << "Enter point and cylinder\n\n";

//inputPoint(x);

inputCylinder(c);

string result = (isInCylinder(x, c)) ? "Yes" : "No";

cout << "Q: Is the point inside of the cylinder?\nA: " << result;

return 0;

}

# 2.

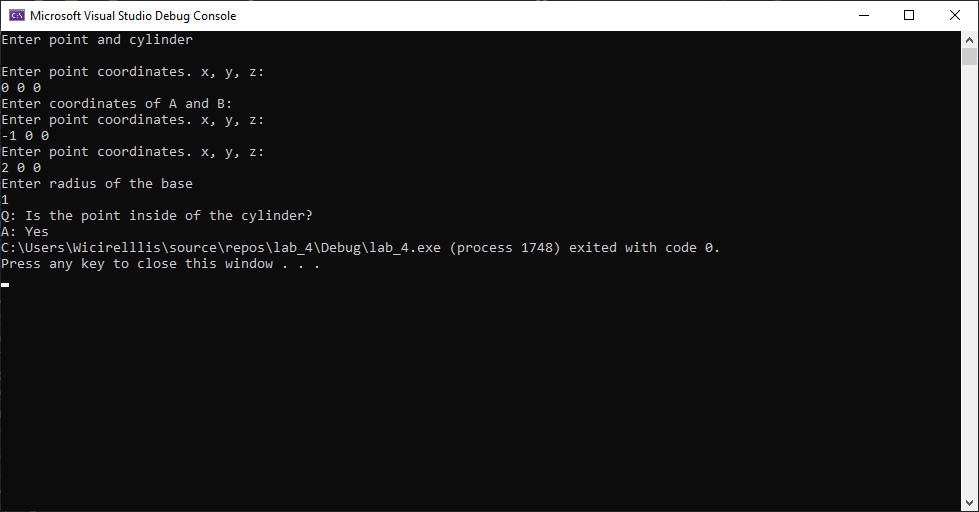
Проверим работу программы на каком-нибудь входе. Пусть ввод слеующий:

0 0 0

-1 0 0

2 0 0

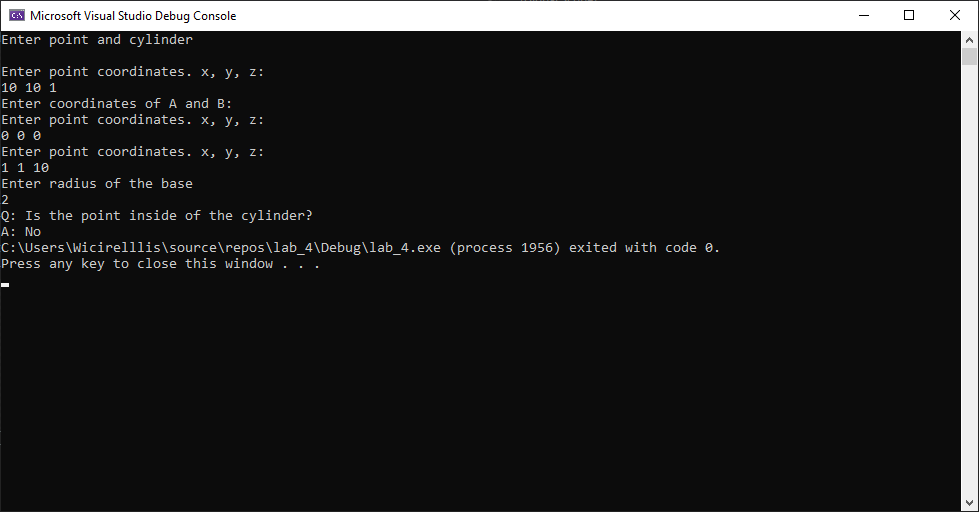
1

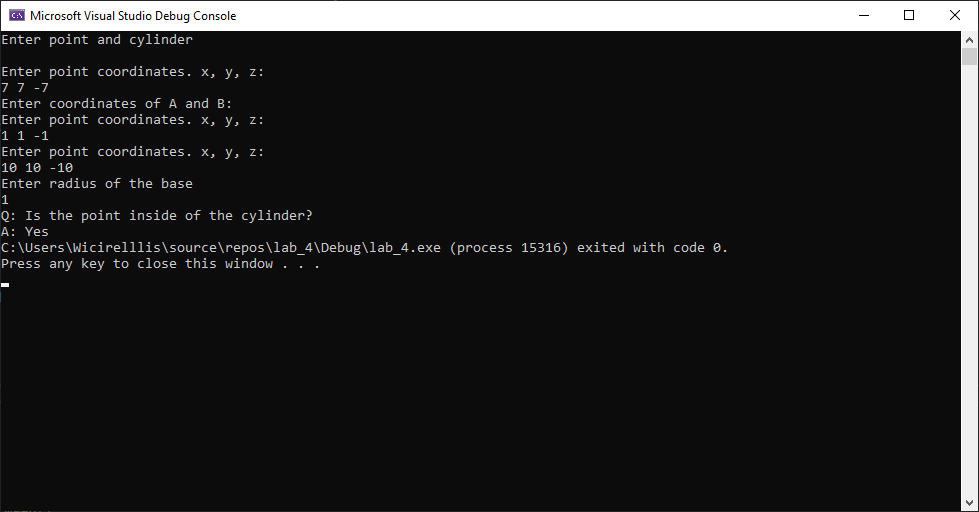


Тогда результат на скрине. Можно понять, что это правда.

# 3.

Несколько скринов вывода





# 4. Вопросы

1) Что такое структура? Какие объекты может содержать структура? Как происходит обращение к элементам структуры?

Структура – это типо пользовательский тип данных, несколько переменных собраных в один объект.

Элементы структуры – это базовые типы (int, double, str итд), другие структуры.

К элементам можно обрщаться через точку и стрелочку. Путь например есть структура, представляющая точку в R3

struct point {

int x;

int y;

int z;

};

point myPoint;

myPoint.x;

point\* ptr\_myPoint;

ptr\_myPoint->x;

myPoint – это инстанс труктуры, и через точку можно получить первую координату. Разница между . и -> в том, что точка применяется к структуре, а стрелочка к указателю на структуру.

2) Возможна ли при инициализации структуры сразу присвоить ей значение? Приведите пример как, если возможно.

Да, возможно. Через равно и фигурные скобки. Например

point myPoint;

myPoint.x = { 0,0,0 };