Make a new directory called **lab12**. Compile and run the programs polyint.cpp presented at the *Curves and Surfaces* section. Change the control points of the program to display the curves that you want.

I modified the control points in the given code and got the first result. After a little more tinkering, I got an almost mirrored image of the same curve.

**First modification:**

```c
GLfloat ctrlpoints[4][3] = {
    { -2.0, -3.0, 0.0}, { -1.0, 2.0, 0.0},
    {1.0, -2.0, 0.0}, {3.0, 3.0, 0.0}};
```

**Second modification:**

```c
GLfloat ctrlpoints[4][3] = {
    { -2.0, -1.0, 0.0}, { -3.0, 1.0, 0.0},
    {2.0, -3.0, 0.0}, {1.0, 1.0, 0.0}};
```
Try to print out \(A\) and its inverse \(A^{-1}\). Try to check if \(A \times A^{-1}\) is the identity matrix.

(Hint: Need to use LinearR4; use LinearMapR4 \(M\) (....) to create the 4x4 matrix; use \(M.Inverse()\) to find inverse; use \(M *= M\_inv\) to find product.)

I was confused with this part at first, since I had no idea what it wanted me to do until I was told to write a c++ file that ran some numbers through the given program. This is what I came up with:

```c++
#include "LinearR4.h"
using namespace std;

int main()
{
    LinearMapR4 x(1, 0, 0, 0,
                  1, 1.0/3.0, 1.0/9.0, 1.0/27.0,
                  1, 2.0/3.0, 4.0/9.0, 8.0/27.0,
                  1, 1, 1, 1);
    LinearMapR4 y;
    y = x.Inverse();
    x *= y;
```
cout << "\nMatrix:\n" << x;
cout << "\nInverse:\n" << y;
cout << "\nProduct and Inverse:\n" << x;
return 0;
}

Summary: After some confusion, I completed all parts of the lab.