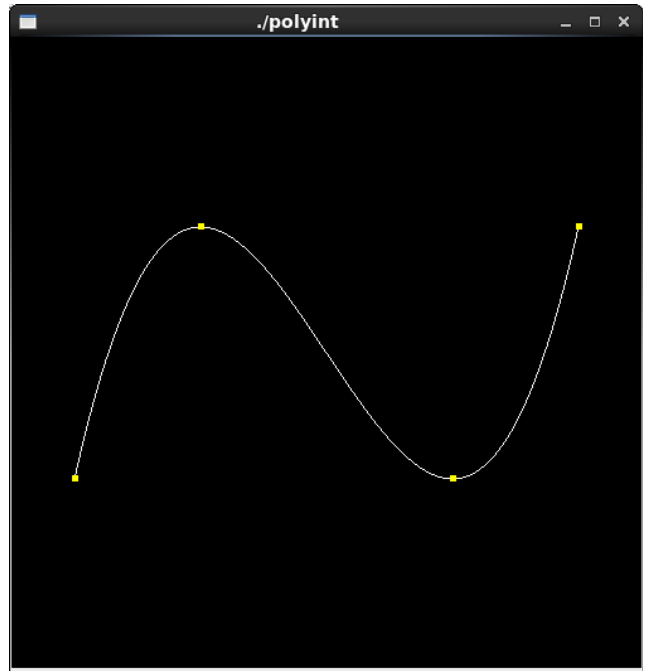
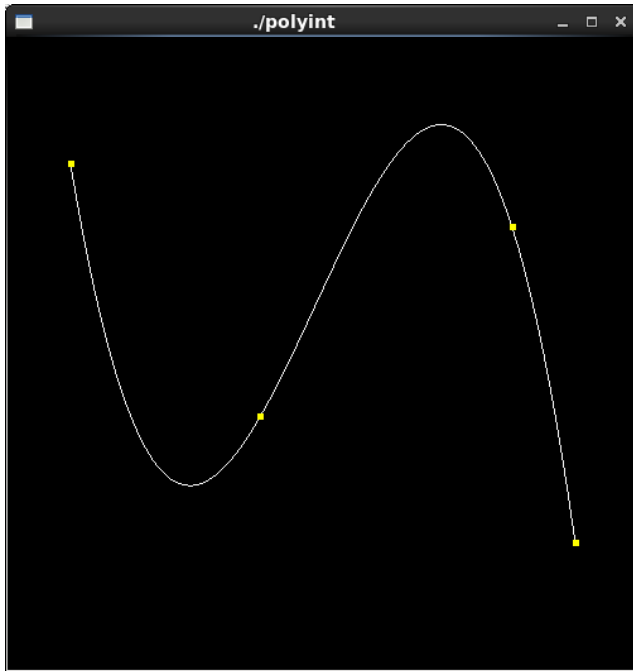


Yazhuo Liu
Lab 12

Compile and run the programs `polyint.cpp`. Change the control points of the program to display the curves that you want.



Code:

```
//polyint.cpp
```

```
...  
GLfloat ctrlpoints[4][3] = {  
    { -4.0, 3.0, 0.0}, { -1.0, -1.0, 0.0},  
    { 3.0, 2.0, 0.0}, { 4.0, -3.0, 0.0}};  
...
```

```
...  
GLfloat ctrlpoints[4][3] = {  
    { -4.0, -2.0, 0.0}, { -2.0, 2.0, 0.0},  
    { 2.0, -2.0, 0.0}, { 4.0, 2.0, 0.0}};  
...
```

Try to print out A and its inverse A^{-1} . Try to check if $A * A^{-1}$ is the identity matrix.

```
004194007@jb358-6:/students/csci/004194007/cse520/lab_12/p2
File Edit View Search Terminal Tabs Help
004194007@jb358-6:/students/csci/00... X 004194007@jb358-6:/students/csci/004... X
[004194007@jb358-6 p2]$ ./display

Matrix A:
(1,      0,      0,      0)
(1,      0.33,    0.11,    0.04)
(1,      0.66,    0.44,    0.3)
(1,      1,      1,      1)

Inverse A:
(1,      -0,      0,      -0)
(-5.54545, 9.09091, -4.54545, 1)
(9.25974, -23.3766, 18.8312, -4.71429)
(-4.71429, 14.2857, -14.2857, 4.71429)

A * Inverse A:
(1,      0,      0,      0)
(5.55112e-17, 1,      -1.11022e-16, 5.55112e-17)
(1.11022e-15, 0,      1,      0)
(2.66454e-15, 0,      -1.77636e-15, 1)

[004194007@jb358-6 p2]$
```

Code:

```
//display.cpp
```

```
...
void print ( LinearMapR4 &M )
{
    cout << "(" << M.m11 << ",\t" << M.m12 << ",\t" << M.m13 << ",\t" << M.m14 << ")" << endl;
    cout << "(" << M.m21 << ",\t" << M.m22 << ",\t" << M.m23 << ",\t" << M.m24 << ")" << endl;
    cout << "(" << M.m31 << ",\t" << M.m32 << ",\t" << M.m33 << ",\t" << M.m34 << ")" << endl;
    cout << "(" << M.m41 << ",\t" << M.m42 << ",\t" << M.m43 << ",\t" << M.m44 << ")" << endl;
}

void display(void)
{
    LinearMapR4 M;
    M.SetColumn1 ( 1, 1, 1, 1 );
    M.SetColumn2 ( 0, 0.33, 0.66, 1 );
```

```
M.SetColumn3 ( 0, 0.11, 0.44, 1 );  
M.SetColumn4 ( 0, 0.04, 0.30, 1 );
```

```
cout << endl << "Matrix A: \n";  
print(M);  
cout << endl << "Inverse A: \n";
```

```
LinearMapR4 N;  
N = M.Inverse();
```

```
print(N);  
cout << endl << "A * Inverse A: \n";
```

```
M *= N;
```

```
print(M);  
cout << endl;
```

```
}  
...
```

Report:

I completed part 1 successfully, but had some difficulties on part 2. I couldn't figure out where I did wrong. The inverse matrix is incorrect, therefore the I couldn't calculate the identity matrix. Since I didn't finish all parts of lab 12, I'm deducting 5 points from my score.