

Московский авиационный институт
(национальный исследовательский университет)

Факультет информационных технологий и прикладной
математики

Кафедра вычислительной математики и программирования

Лабораторная работа №1 по курсу «Компьютерная графика»

Студент: М. А. Бронников
Преподаватель: Г. С. Филиппов
Группа: М8О-307Б
Дата:
Оценка:
Подпись:

Москва, 2019

Построение изображений 2D- кривых

Задача: Написать и отладить программу, строящую изображение заданной замечательной кривой. Обеспечить автоматическое масштабирование и центрирование кривой при изменении размеров окна.

Вариант №12:

Кривая: $p = a \cos 3\varphi$

p, φ - полярные координаты

a - константа, значение которой вводится с клавиатуры

1 Описание

Для реализации я использовал библиотеку GLUT и язык C++.

Идея заключается в последовательной отрисовке точек графика с заданной частотой аппроксимации, при малых значениях которой график на картинке достаточно приближен к идеальной математической модели.

График изображается в декартовых координатах, поэтому приходится реализовывать функцию преобразования из полярных координат в декартовы, после чего библиотека сама переведет эти координаты в экранные, а затем в пиксели.

Корректное отображение при изменении размеров окна также реализуется библиотекой, которую нам потребуется лишь инициализировать в начале программы.

2 Исходный код

Вначале подключаются необходимые. Библиотека GLUT подключается как «GL/glut.h»

```
1 || #include <GL/glut.h>
```

Затем задаются глобальные переменные от которых зависит вид графика и его аппроксимация.

```
1 || int X_COORD;  
2 || int Y_COORD;  
3 || const float ITERATIONS = 0.00005;  
4 || int x_off;  
5 || int y_off;  
6 || float a;
```

Опишем функции которые будут принимать на вход полярные координаты и возвращать декартовы для отрисовки:

```
1 || float x_from_polar(float r, float f){
2 ||     return r*cos(f);
3 || }
4 || float y_from_polar(float r, float f){
5 ||     return r*sin(f);
6 || }
7 || float r_function(float f){
8 ||     return a*cos(3*f);
9 || }
```

Функция *drawgrid* отрисовывает координатную прямую с отметками на ней.

```
1 || void drawgrid(float SERIF_OFFSET, float SERIF_DISTANCE);
```

Самой отрисовкой графика занимается функция *drawfunc*, отрисовывающая график на $[0, 2\pi]$ вызывая в себе функции преобразования из полярных координат в декартовы.

```
1 || void drawfunc();
```

Отрисовка происходит в бесконечном «игровом» цикле, который выполняет в себе функцию *dispay* а также обрабатывает обратные вызовы.

```
1 || void display();
```

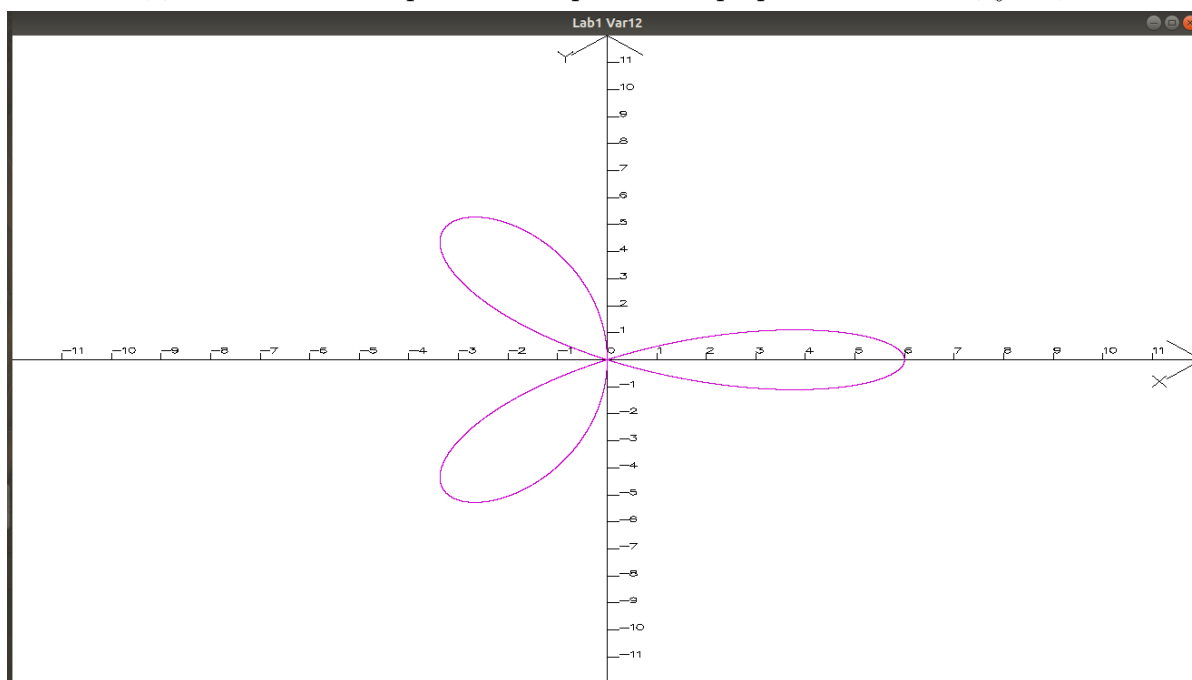
В функции *main* происходит начальная инициализация библиотеки для корректного отображения.

3 Консоль

В консоли необходимо скомпилировать исходный код и запустить. В окне необходимо будет ввести параметр *a*, от которого зависит вид графика.

```
(base) max@max-X550CC:~/KG/lab1$ g++ main.cpp -o main -lglut -lGL -lGLU
(base) max@max-X550CC:~/KG/lab1$ ./main
Hi! Program made by Maksim Bronnikov(M80-307)!
Enter a: 10.5
(base) max@max-X550CC:~/KG/lab1$ exit
```

После ввода значения a откроется изображение графика в окне следующего типа:



Это окно можно изменять по размерам и перемещать по экрану без всяких побочных эффектов, график подстраивается под изменение размеров экрана и масштабируется соответствующим образом.

4 Выводы

Выполнив первую лабораторную работу по курсу «Компьютерная графика», я получил представление о том, что такое компьютерная графика и как она реализуется на языке C++ посредством библиотеки GLUT.

В процессе решения задачи я столкнулся с такими трудностями, как изображение кривой имея в арсенале лишь возможность отрисовки прямой по точкам и отрисовка самой точки на графике. Помимо этого мне пришлось столкнуться с проблемой перевода полярных координат в интерпретируемые библиотекой для последующей отрисовки на экране.

Я думаю что данная лабораторная работа послужила доступным введением в увлекательный мир компьютерной графики.