

Федеральное агентство связи
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Сибирский государственный университет телекоммуникаций и
информатики»

Лабораторная работа №2
«Процессы и асинхронное взаимодействие. Часть 1, 2»

Выполнил: студент 4 курса

ИВТ, гр. ИП-713

Михеев Н.А.

Проверил: старший преподаватель кафедры ПМиК

Милешко А.В.

Новосибирск, 2020 г.

Задание

1. Тщательно изучить библиотеку VinGraph.
2. Используя функции библиотеки VinGraph, нарисовать абстрактную картину, которой представлены (почти) все доступные графические элементы.

Выполнение задания

Была изучена библиотека VinGraph по соответствующей методичке. Далее была разработана программа с использованием библиотеки `vingraph.h` по отрисовке всех доступных графических элементов.

Из-за ошибки в реализации дуги (Arc) после ее отрисовки происходит сбой в дальнейшей отрисовке и дальнейшее мгновенное завершение программы. Поэтому сначала были отрисованы все элементы без Arc и затем отдельно сама дуга с 5 секундной задержкой после ее отрисовки.

На работу программы выделено 10 секунд, после которых происходит вызов метода `CloseGraph()` и программа закрывается.

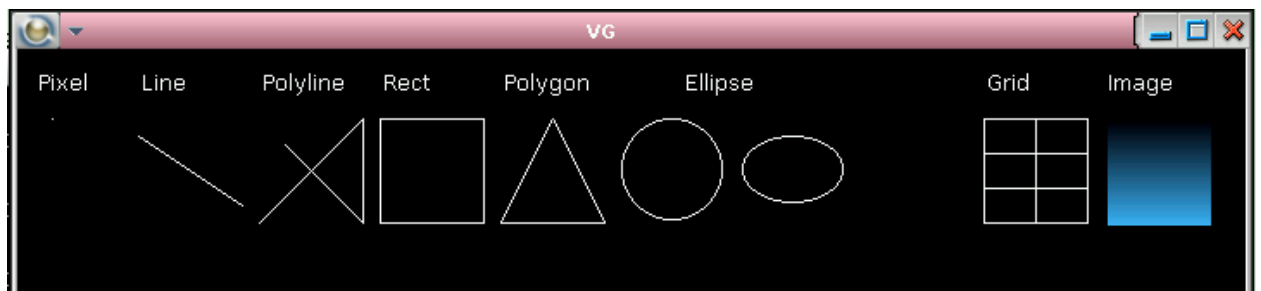


Рис.1 – графическое окно со всеми фигурами

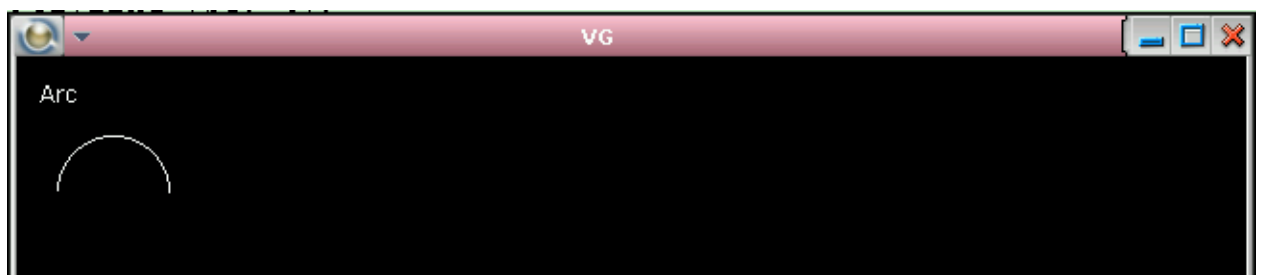


Рис.2 – графическое окно с отдельной отрисовкой дуги (Arc)

Листинг программы

```
#include <stdlib.h>
#include <unistd.h>
#include <vingraph.h>

int main()
```

```

{
    ConnectGraph();
    Text(10, 10, "Pixel");
    Pixel(20, 40);

    Text (70, 10, "Line");
    Line (70, 50, 130, 90);

    Text (140, 10, "Polyline");
    tPoint p1 [] = { {140, 100}, {200, 40}, {200, 100}, {155, 55} };
    Polyline (p1, 4);

    Text (210, 10, "Rect");
    Rect (210, 40, 60, 60);

    Text (280, 10, "Polygon");
    tPoint p2 [] = { {280, 100}, {310, 40}, {340, 100} };
    Polygon (p2, 3);

    Text (385, 10, "Ellipse");
    Ellipse (350, 40, 60, 60);
    Ellipse (420, 50, 60, 40);

    //Text (10, 10, "Arc");
    //Arc (20, 40, 60, 60, 0, 1800);
    //sleep(10);

    Text (560, 10, "Grid");
    Grid (560, 40, 60, 60, 3, 2);

    int *im_buf = (int*) malloc (60*60*4);
    for (int i = 0, c = 10; i < 60; i++, c += 0x010304)
        for (int j = 0; j < 60; j++)
            im_buf [60*i + j] = c;
    Text (630, 10, "Image");
    Image32 (630, 40, 60, 60, im_buf);

    sleep(10);
    CloseGraph();
    return 0;
}

```