**Лабораторная работа №6**

**Графика**

**Задание 1.** Нарисовать график функции y=x\*x

**Код программы**

#include <graphics.h> // включить описание графических функций С/С++

#include <conio.h>

int main()

{

float x, y;

/\* Автоматическое определение графических параметров \*/

int gdriver = DETECT, gmode;

initgraph(&gdriver, &gmode, ""); // Инициализация графического режима

x=-10; //Инициализируем x

moveto(x,x\*x); //Устанавливаем курсор

do

{

y=x\*x; //Наша формула для построения графика

lineto(x\*50+getmaxx()/2, getmaxy()/2-(y\*20)); // ..откуда рисуем график

x=x+0.02;

}

while(x<10);

getch(); // ожидание нажатия пользователем любой клавиши

closegraph(); // выход из графического режима

return 0;

}



**Задание 2.** Написать программу, которая использует стандартные функции для рисования на белом фоне красного прямоугольника с синей границей и синими диагоналями, и желтого круга с фиолетовой границей в центре экрана.

**Код программы**

#include <graphics.h>

#include <conio.h>

int main()

{

int d = DETECT, m;

initgraph (&d, &m, "C:\\BORLANDC\\BGI");

setfillstyle (SOLID\_FILL, WHITE);

bar (0, 0, 639, 479);

setfillstyle ( SOLID\_FILL, RED);

bar (220, 160, 420, 320);

setcolor ( BLUE );

rectangle (220, 160, 420, 320);

line (220, 160, 420, 320);

line (220, 320, 420, 160);

setcolor (MAGENTA);

circle (320, 240, 30);

setfillstyle (SOLID\_FILL, YELLOW);

floodfill (300, 250, MAGENTA);

setcolor (GREEN);

getch();

closegraph();

}



**Задание 3.** Построить график циклоиды***.***

Математическое описание:x = a t - b sin t, y = a - b cos t,

при b<a циклоида получается укороченной, при b>a - удлиненной, b = a дает обычную циклоиду.

**Код программы**

#include <graphics.h>

#include <conio.h>

#include <math.h>

#define M\_PI 3.1415

int main()

{

int gd = DETECT, gm, k;

float a, b, t;

initgraph(&gd,&gm,0);

setcolor(1);

moveto(0, 250);

linerel(625, 0);

setcolor(3);

line(10, 0, 10, 478);

setcolor(5);

moveto(10, 240);

t=0.0; // текущее значение параметра t из задания

for ( k=0; k<200; ++k )

{

a = 0.4 \* t - 0.5\*sin(t);

b = 0.4 - 0.5\*cos(t);

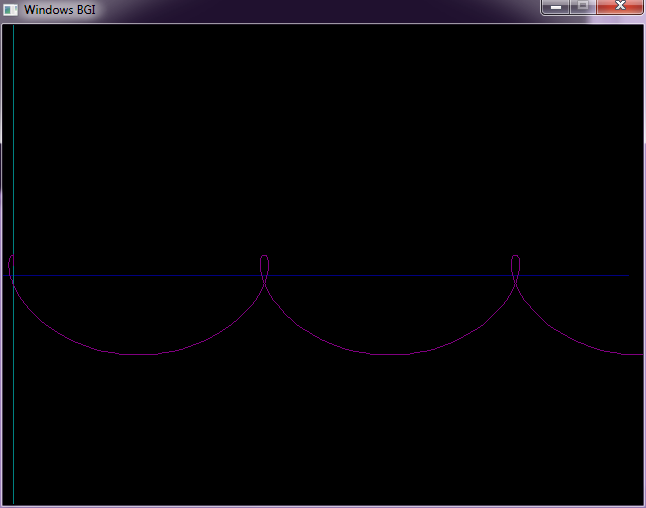
lineto(100\*a + 10, 100\*b + 240);

t+=M\_PI/32;

}

getch();

}



**Задание 4**. Подготовить и отладить программу вычерчивания квадрата, вписанного в окружность. Запрограммировать изменение цветов окружности.

**Код программы**

#include "stdio.h"

#include "graphics.h"

#include "math.h"

int main()

{

int gd = DETECT, gm; initgraph(&gd, &gm, 0);

int srx, sry, x,y,r,i;

srx=(getmaxx()+1)/2;

sry=(getmaxy()+1)/2;

printf("%d,%d\n",srx,sry);

printf("Vvedite koordinaty x \n");

scanf("%d",&x);

printf("Vvedite koordinaty y \n");

scanf("%d",&y);

printf("Vvedite radiys okryzhnosti \n");

scanf("%d",&r);

for (i=1; i<=15; i++)

{

setcolor(i);

setfillstyle(1,i);

fillellipse(x,y,r,r);

setfillstyle(SOLID\_FILL,4);

setcolor(4);

bar(x-r/sqrt(2),y-r/sqrt(2),x+r/sqrt(2),y+r/sqrt(2));

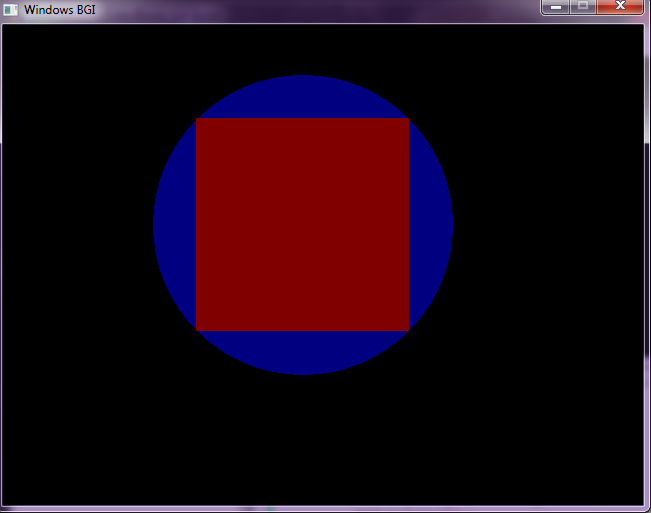
getch();

}

getch();

return 0;

}



**Задание 5.** Разработать программу изображения точки, движущейся по косинусоиде (y = cos(x)).

**Код программы**

#include "stdio.h"

#include "graphics.h"

#include "math.h"

int main()

{

int gd = DETECT, gm; initgraph(&gd, &gm, 0);

setbkcolor(15);

int midx, midy, x, y;

midx=(getmaxx()+1)/2;

midy=(getmaxy()+1)/2;

printf("%d,%d",midx,midy);

line(0,midy,getmaxx(),midy);

line(midx,0,midx,getmaxy());

getchar();

for (x=-70; x<=360; x++)

{

putpixel(x\*10+320,-(cos(x))\*20+240,15);

Sleep(60);

putpixel(x\*10+320,-(cos(x))\*20+240,0);

line(0,midy,getmaxx(),midy);

line(midx,0,midx,getmaxy());

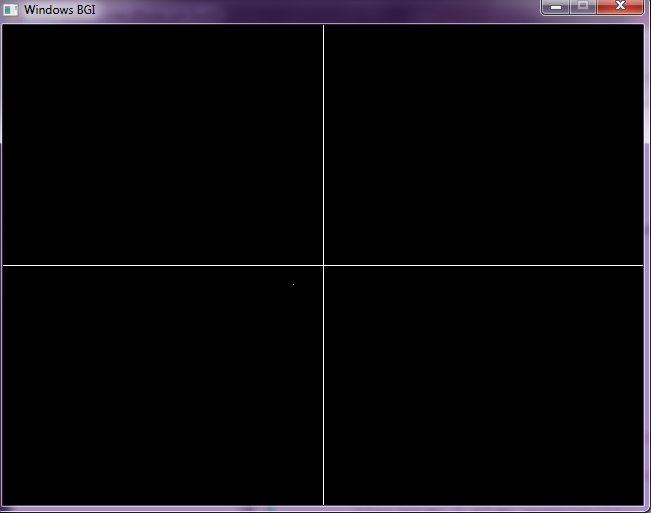
}

getch();

return 0;

}





**Задание 6.** Изобразить движущийся на зрителя экран (прямоугольник).

**Код программы**

#include "stdio.h"

#include "graphics.h"

#include "math.h"

int main()

{

int gd = DETECT, gm; initgraph(&gd, &gm, 0);

setbkcolor(15);

int midx, midy, x, y;

midx=(getmaxx()+1)/2;

midy=(getmaxy()+1)/2;

printf("%d,%d",midx,midy);

line(0,midy,getmaxx(),midy);

line(midx,0,midx,getmaxy());

getchar();

for (x=-70; x<=360; x++)

{

putpixel(x\*10+320,-(cos(x))\*20+240,15);

Sleep(90);

putpixel(x\*10+320,-(cos(x))\*20+240,0);

line(0,midy,getmaxx(),midy);

line(midx,0,midx,getmaxy());

}

getch();

return 0;

}

