

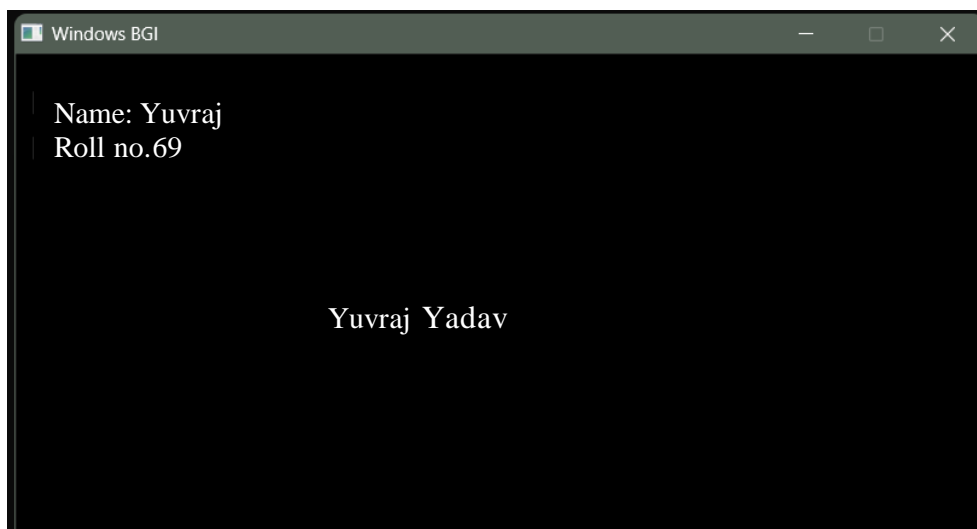
Practical 8

8 Solve the following:

a. Develop a simple text screen saver using graphics functions.

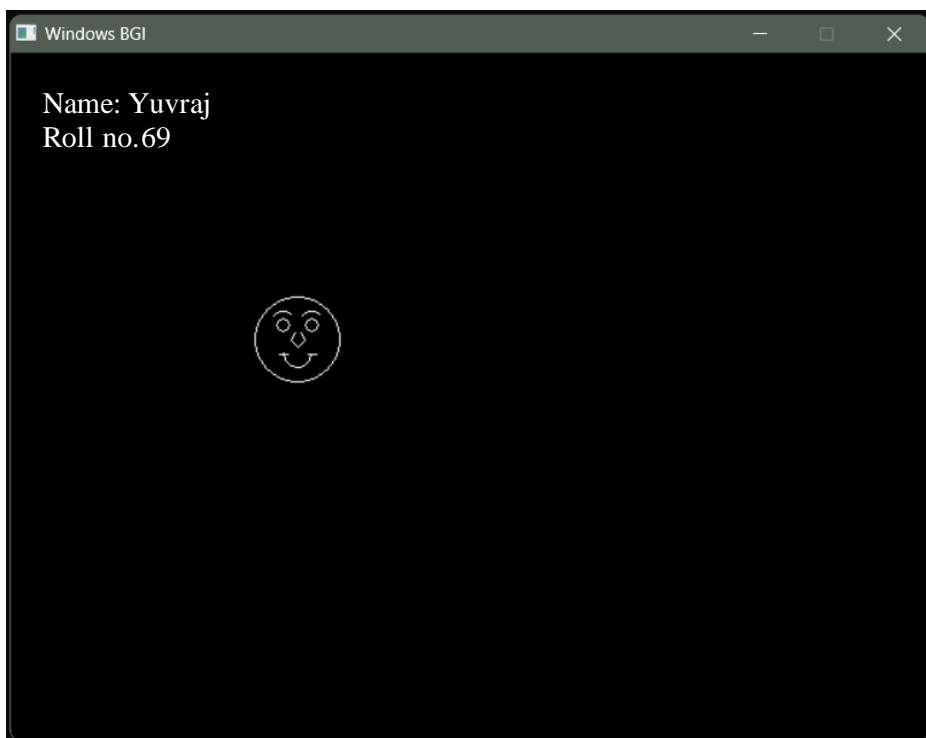
```
#include<graphics.h>
#include<conio.h>
#include<stdio.h>
#include<dos.h>
int main()
{
int gd=DETECT,gm,i,maxx,maxy,key0;
initgraph(&gd,&gm,"C:\\TC\\BGI");
maxx=getmaxx();
maxy=getmaxy();
while(!kbhit())
{
for(i=0;i<maxy;i++)
{
cleardevice();
settextstyle(3,0,3);
outtextxy(10,20,"Name: Yuvraj ");
outtextxy(10,50,"Roll.No: 69 ");
outtextxy(maxy/2,i,"Yuvraj Yadav");
delay(10);
}
}
getch();
closegraph();
return 0;
}
```

Output:



b. Perform smiling face animation using graphic functions.

```
#include<graphics.h>
#include<conio.h>
#include<stdio.h>
int main()
{
    int gd=DETECT,gm;
    initgraph(&gd,&gm,"C:\\TC\\BGI");
    circle(200,200,30);
    circle(190,190,5);
    arc(190,190,50,130,10);
    circle(210,190,5);
    arc(210,190,50,130,10);
    arc(200,210,180,360,10);
    line(187,210,193,210);
    line(207,210,213,210);
    line(198,195,195,200);
    line(202,195,205,200);
    line(195,200,200,205);
    line(205,200,200,205);
    outtextxy(10,20,"Name: Yuvraj");
    outtextxy(10,50,"Roll no. 69");
    getch();
    closegraph();
}
```

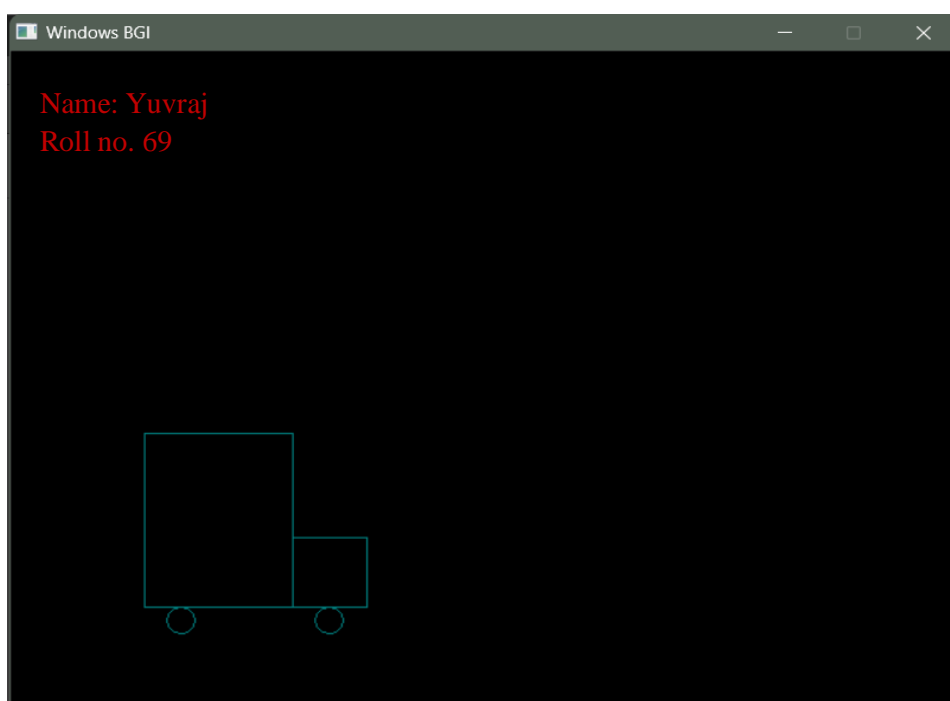
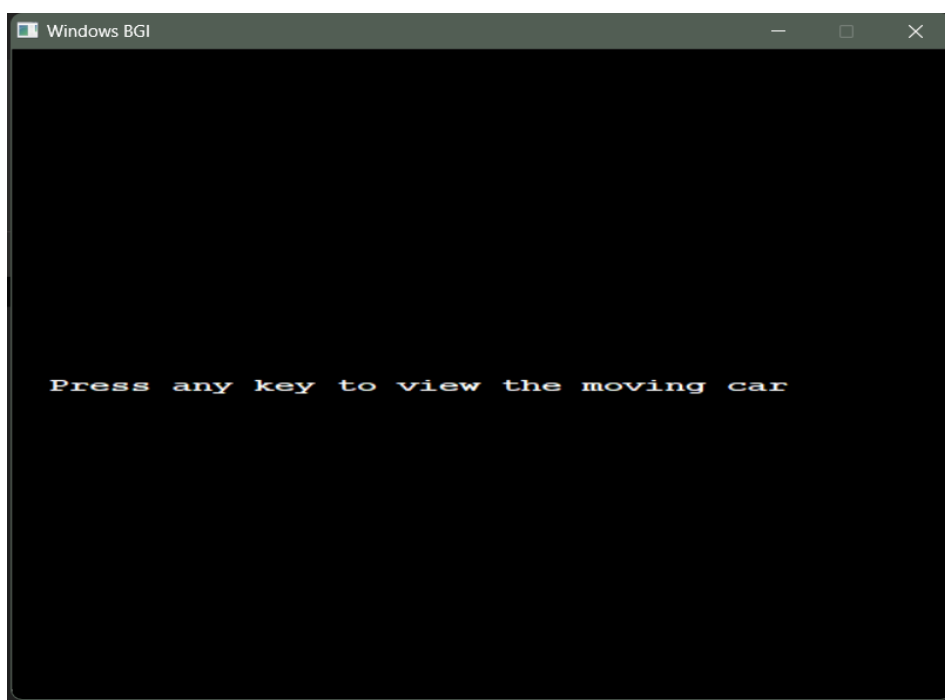
Output:

c. Draw the moving car on the screen.

```
#include<graphics.h>
#include<dos.h>
#include<conio.h>
int main()
{
    int i,j=0,gd=DETECT,gm;
    initgraph(&gd,&gm,"C:\\TC\\BGI");
    settextstyle(DEFAULT_FONT,HORIZ_DIR,2);
    outtextxy(25,240,"Press any key to view the moving car");
    getch();
    setviewport(0,0,639,440,1);
    for(i=0;i<=420;i=i+10,j++)
    {
        rectangle(50+i,275,150+i,400);
        rectangle(150+i,350,200+i,400);
        circle(75+i,410,10);
        circle(175+i,410,10);
        setcolor(j);
        outtextxy(10,20,"Name: Yuvraj");
        outtextxy(10,50,"Roll no. 69");
        delay(1000);
        clearviewport();

    }
    getch();
    return 0;
}
```

Output:

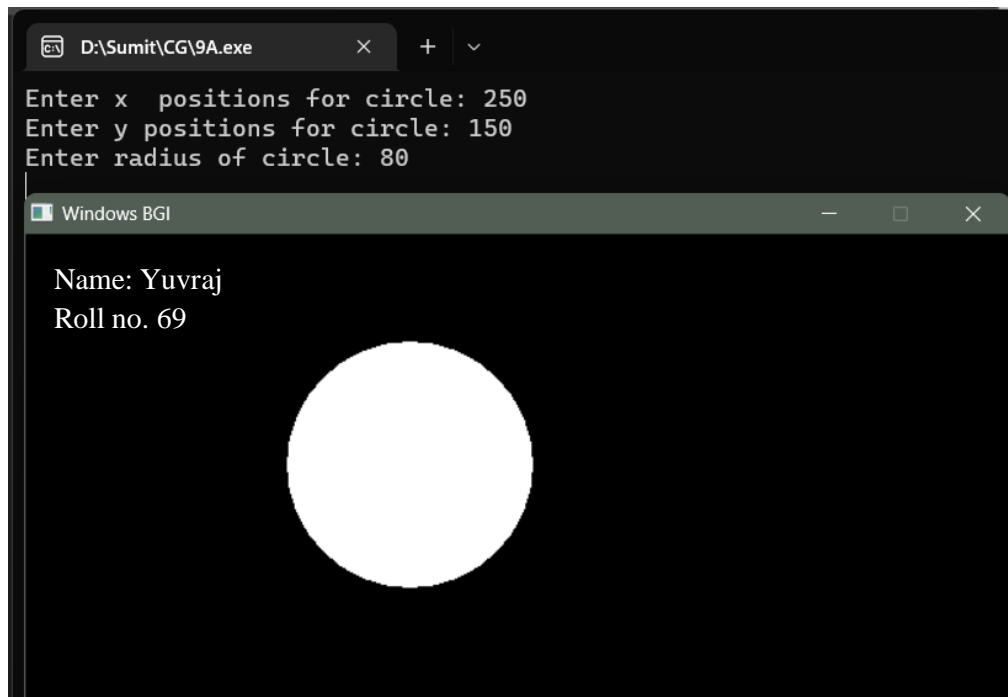


Practical 9

9 Solve the following:

a. Write a program to fill a circle using Flood Fill Algorithm.

```
#include<stdio.h>
#include<graphics.h>
#include<dos.h>
int floodFill(int x,int y,int oldcolor,int newcolor)
{
if(getpixel(x,y) == oldcolor)
{
putpixel(x,y,newcolor);
floodFill(x+1,y,oldcolor,newcolor);
floodFill(x,y+1,oldcolor,newcolor);
floodFill(x-1,y,oldcolor,newcolor);
floodFill(x,y-1,oldcolor,newcolor);
}
}
//getpixel(x,y) gives the color of specified pixel
int main()
{
int gm,gd=DETECT,radius;
int x,y;
printf("Enter x positions for circle: ");
scanf("%d",&x);
printf("Enter y positions for circle: ");
scanf("%d",&y);
printf("Enter radius of circle: ");
scanf("%d",&radius);
initgraph(&gd,&gm,"c:\\turbo3\\bgi");
circle(x,y,radius);
floodFill(x,y,0,15);
delay(1000);
outtextxy(10,20,"Name: Yuvraj");
outtextxy(10,50,"Roll no. 69");
getch();
closegraph();
return 0;
}
```

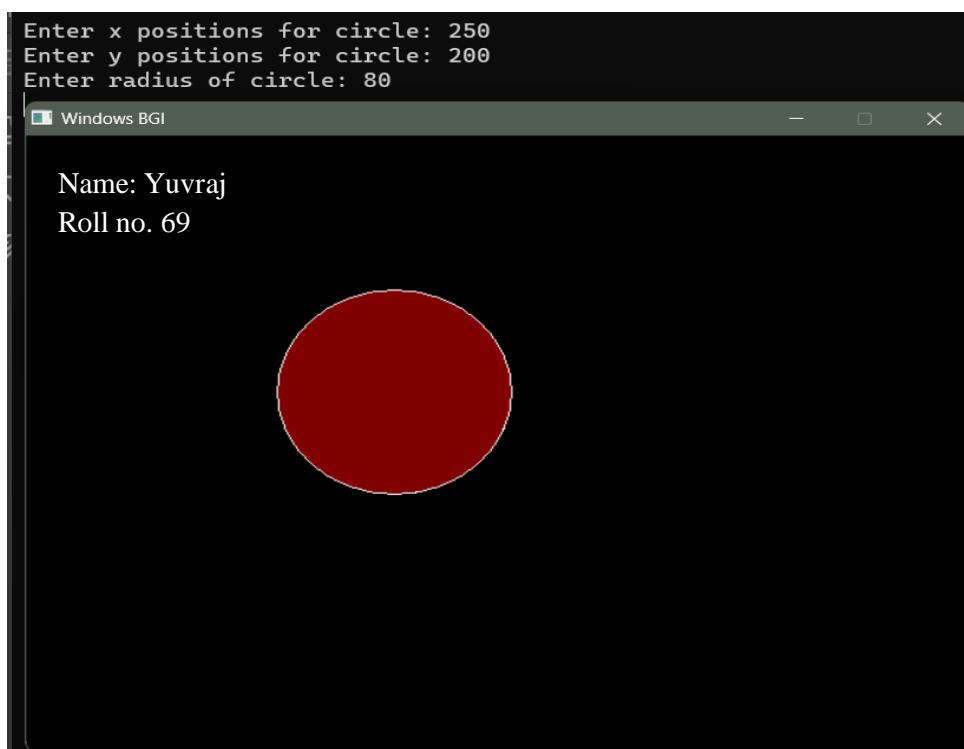
Output:**b. Write a program to fill a circle using Boundary Flood Algorithm.**

```
#include<stdio.h>
#include<graphics.h>
#include<dos.h>
void boundaryfill(int x,int y,int f_color,int b_color)
{
if(getpixel(x,y)!=b_color && getpixel(x,y)!=f_color)
{
putpixel(x,y,f_color);
boundaryfill(x+1,y,f_color,b_color);
boundaryfill(x,y+1,f_color,b_color);
boundaryfill(x-1,y,f_color,b_color);
boundaryfill(x,y-1,f_color,b_color);
}
}
//getpixel(x,y) gives the color of specified pixel
int main()
{
int gm,gd=DETECT,radius;
int x,y;
printf("Enter x positions for circle: ");
```

```
scanf("%d",&x);
printf("Enter y positions for circle: ");
scanf("%d",&y);

printf("Enter radius of circle: ");
scanf("%d",&radius);
initgraph(&gd,&gm,"c:\\turbo3\\bgi");
circle(x,y,radius);
boundaryfill(x,y,4,15);
outtextxy(10,20,"Name: Yuvraj");
outtextxy(10,50,"Roll no. 69");
delay(5000);
closegraph(20);
return 0;
}
```

Output:



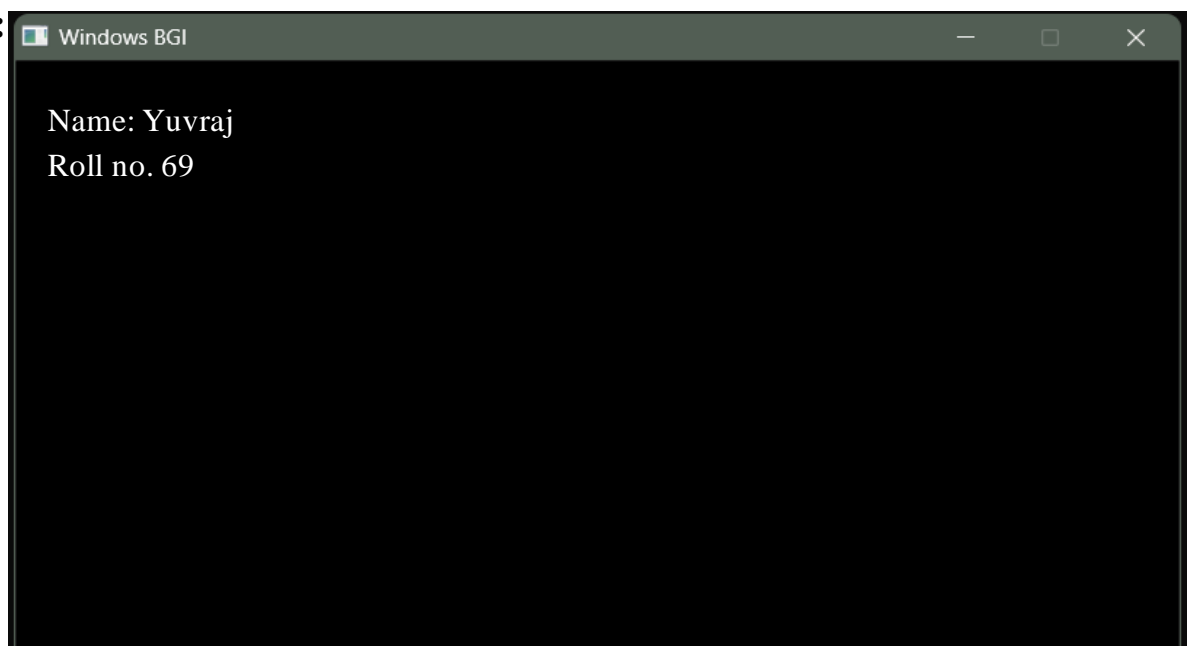
Practical 10

10 Solve the following:

a. Develop a simple text screen saver using graphics functions.

```
#include<stdio.h>
#include<conio.h>
#include<graphics.h>
int main()
{
int grd=DETECT,grm,i,maxx,maxy,key = 0;
initgraph(&grd,&grm,"c://turboc3//bgi");
maxx=getmaxx();/* Getting coordinates of the monitor */
maxy=getmaxy();
while(!kbhit())
{
for(i=0;i<maxy;i++)
{
cleardevice();
settextstyle(1,0,2); //font direction size
outtextxy(10,20,"Name: Yuvraj");
outtextxy(10,50,"Roll no. 69");
delay(30);
}
}
closegraph();
return 0;
}
```

Output:



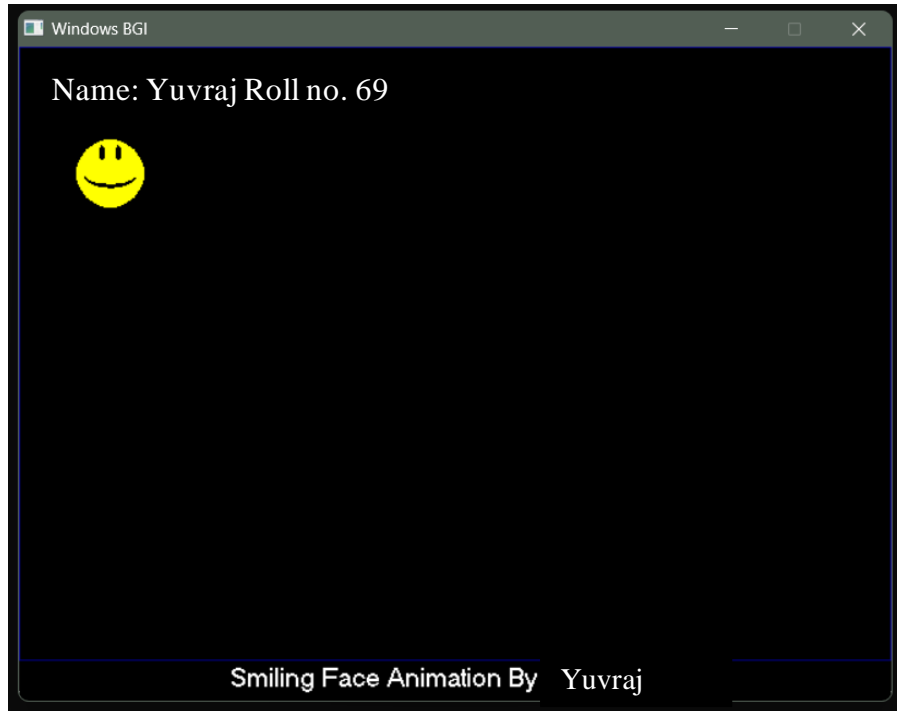
b. Perform smiling face animation using graphic functions.

```

#include<graphics.h>
#include<conio.h>
#include<stdlib.h>
#include<dos.h>
main()
{
int gd=DETECT,gm,area,temp1,temp2,left=25,top=75;
void*p;
initgraph(&gd,&gm,"c:\\turbo3\\bgi");
setcolor(YELLOW);
circle(50,100,25);
setfillstyle(SOLID_FILL,YELLOW);
floodfill(50,100,YELLOW);
setcolor(BLACK);
setfillstyle(SOLID_FILL,BLACK);
fillellipse(44,85,2,6);
fillellipse(56,85,2,6);
ellipse(50,100,205,335,20,9);
ellipse(50,100,205,335,20,10);
ellipse(50,100,205,335,20,11);
area=imagesize(left,top,left+100,top+100);
p=malloc(area);
setcolor(WHITE);
settextstyle(SANS_SERIF_FONT,HORIZ_DIR,2);
outtextxy(155,450,"Smiling Face Animation By Yuvraj");
outtextxy(10,20,"Name: Yuvraj Roll no. 69");
setcolor(BLUE);
rectangle(0,0,639,449);
//while(!kbhit())
{
temp1=1+rand()%600;
temp2=1+rand()%400;
getimage(left,top,left+50,top+60,p);
putimage(left,top,p,XOR_PUT);
putimage(temp1,temp2,p,XOR_PUT);
delay(100);
left=temp1;
top=temp2;
}
}

```

```
getch();  
closegraph();  
return 0;  
}
```

Output:

c. Perform smiling face animation using graphic functions.

```
#include<graphics.h>
#include<dos.h>
#include<conio.h>
main()
{
int i,j=0, gd=DETECT,gm;
initgraph(&gd,&gm,"..\\BGI");
settextstyle(DEFAULT_FONT,HORIZ_DIR,2);
outtextxy(25,240,"Press any key to view the moving car");
getch();
setviewport(0,0,639,440,1);
for(i=0; i<=420;i=i+10,j++)
{
rectangle(50+i,275,150+i,400);
rectangle(150+i,350,200+i,400);
circle(75+i,410,10);
circle(175+i,410,10);
setcolor(YELLOW);
delay(100);
if(i==420)
break;
clearviewport();
}
setcolor(WHITE);
outtextxy(10,20,"Name: Yuvraj Roll no. 69");
getch();
closegraph();
return 0;
}
```

Output:

