10. Solve the following

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

b. Perform smiling face animation using graphic functions.

c. Perform Sad face animation using graphic functions.

d. Draw the moving car on the screen

e. Perform Bouncing Ball animation using graphic function.

Solution :-

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

```
#include <stdio.h>
#include <stdlib.h>
#include <graphics.h>
#include <conio.h>
void main(){
int gdriver=DETECT,gmode,col=480,row=640,font=4,direction=2,size=8,color=15;
initgraph(&gdriver,&gmode,"C:\\TurboC3\\BGI");
cleardevice();
while(!kbhit()){
settextstyle(random(font),random(direction),random(size));
setcolor(random(color));
outtextxy(random(col),random(row),"SYBSC-IT-CGA");
delay(250);
}
closegraph();
}
```

b. Perform smiling face animation using graphic functions.

```
#include <conio.h>
#include <dos.h>
#include <graphics.h>
#include <stdio.h>
// Driver Code
int main()
{

// Initialize graphic driver
int gr = DETECT, gm;
// Initialize graphics mode by passing
// three arguments to initgraph function
// &gdriver is the address of gdriver
// variable, &gmode is the address of
// gmode and "C:\\Turboc3\\BGI" is the
// directory path where BGI files
// are stored
initgraph(&gr, &gm, "C:\\Turboc3\\BGI");
// Set color of smiley to yellow
setcolor(YELLOW);
// creating circle and fill it with
// yellow color using floodfill.
circle(300, 100, 40);
setfillstyle(SOLID_FILL, YELLOW);
floodfill(300, 100, YELLOW);
// Set color of background to black
setcolor(BLACK);
setfillstyle(SOLID_FILL, BLACK);
// Use fill ellipse for creating eyes
fillellipse(310, 85, 2, 6);
```

```
fillellipse(290, 85, 2, 6);
// Use ellipse for creating mouth
ellipse(300, 100, 205, 335, 20, 9);
//ellipse(300, 100, 205, 335, 20, 10);
//ellipse(300, 100, 205, 335, 20, 11);
getch();
// closegraph function closes the
// graphics mode and deallocates
// all memory allocated by
// graphics system

closegraph();
return 0;
}
```

c. Perform Sad face animation using graphic functions.

d. Draw the moving car on the screen

```
#include <stdio.h>
#include <graphics.h>
#include <conio.h>
#include <dos.h>
int main() {
int gd = DETECT, gm;
int i, maxx, midy;
/* initialize graphic mode */
initgraph(&gd, &gm, "X:\\TC\\BGI");
```

```
/* maximum pixel in horizontal axis */
maxx = getmaxx();
/* mid pixel in vertical axis */
midy = getmaxy()/2;
for (i=0; i < maxx-150; i=i+5) {
/* clears screen */
cleardevice();
/* draw a white road */
setcolor(WHITE);
line(0, midy + 37, maxx, midy + 37);
/* Draw Car */
setcolor(YELLOW);
setfillstyle(SOLID_FILL, RED);
line(i, midy + 23, i, midy);
line(i, midy, 40 + i, midy - 20);
line(40 + i, midy - 20, 80 + i, midy - 20);
line(80 + i, midy - 20, 100 + i, midy);

line(100 + i, midy, 120 + i, midy);
line(120 + i, midy, 120 + i, midy + 23);
line(0 + i, midy + 23, 18 + i, midy + 23);
arc(30 + i, midy + 23, 0, 180, 12);
line(42 + i, midy + 23, 78 + i, midy + 23);
arc(90 + i, midy + 23, 0, 180, 12);
line(102 + i, midy + 23, 120 + i, midy + 23);
line(28 + i, midy, 43 + i, midy - 15);
line(43 + i, midy - 15, 57 + i, midy - 15);
line(57 + i, midy - 15, 57 + i, midy);
line(57 + i, midy, 28 + i, midy);
line(62 + i, midy - 15, 77 + i, midy - 15);
line(77 + i, midy - 15, 92 + i, midy);
```

```
line(92 + i, midy, 62 + i, midy);
line(62 + i, midy, 62 + i, midy - 15);
floodfill(5 + i, midy + 22, YELLOW);
setcolor(BLUE);
setfillstyle(SOLID_FILL, DARKGRAY);
/* Draw Wheels */
circle(30 + i, midy + 25, 9);
circle(90 + i, midy + 25, 9);
floodfill(30 + i, midy + 25, BLUE);
floodfill(90 + i, midy + 25, BLUE);
/* Add delay of 0.1 milli seconds */
delay(100);
}
getch();
closegraph();
return 0;
}
```

e. Perform Bouncing Ball animation using graphic function.

```
#include <stdio.h>
#include <conio.h>
#include <graphics.h>
#include <dos.h>
int main() {
int gd = DETECT, gm;

int i, x, y, flag=0;
initgraph(&gd, &gm, "C:\\TC\\BGI");
/* get mid positions in x and y-axis */
```

```
x = getmaxx()/2;
y = 30;

while (!kbhit()) {
if(y >= getmaxy()-30 || y <= 30)
flag = !flag;
/* draws the gray board */
setcolor(RED);
setfillstyle(SOLID_FILL, RED);
circle(x, y, 30);
floodfill(x, y, RED);
/* delay for 50 milli seconds */
delay(50);
/* clears screen */
cleardevice();
if(flag){
y = y + 5;
} else {
y = y - 5;
}
}
getch();
closegraph();
return 0;
}
```