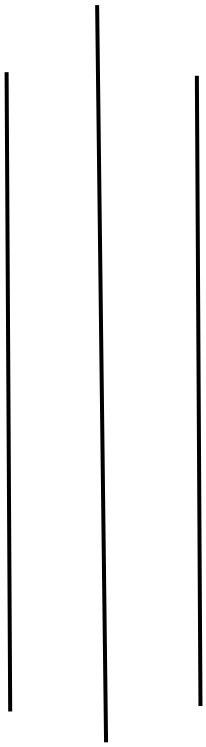
C PROGRAMMING PROJECT ON SNAKE GAME

TRIBHUVAN UNIVERSITY,

PULCHWOK CAMPUS



PREPARED BY: PROJECT TEAM:

ANJALI RANA MAGAR (077BEL05)

SAURAV MAHARJAN (077BEL041)

RAJU GUPTA (077BEL031)

SANDIP ROKAYA (077BEL037)

SUBMITTED TO:

THE DEPARTMENT OF ELECTRONIC AND COMPUTER DEPARTMENT

BACKGROUND:

After learning the C programming languge, our team

Moved to know the decent application of the widely used fundamental programming language(*c program*).

In this game we have implemented the following concept in c programming

1.basic structure of c.

2.user defined functions.

3.structures.

4.file handling.

5.complex loops.

6.string.

7.array and pointers.

8.control statements.

About the snake game:

Talking about the feature and the accessories of the game simply, player have to enter his name where the score will saved in that name. then the game starts with

the rectangular interface made by character containing head of snake in middle and a fruit on random place inside the border. ‘w’,’q’,’e’ and ’d’ are used to move the snake head. As the snake head touches the fruit. Another random fruit is generated and correspondingly the tail is increased by a character increasing the length of the snake. When the game reach certain score game is quit and level 2 starts with more complexity than level 1. And so on. Finally score is saved in the name entered previously.

When the game is played by many users, it compare the score among the players in the leaderboard showing score obtained by the users.

General Functions used:

Setup ():

it is userdefined function to create all the graphical structures seen in interface like snake, fruit, borders etc.

Fruit ():

It is function to generate the fruit randomly inside the border using rand() function.

snake():

it locates the position of snake inside the border.

Move():

It moves the snake head and the corresponding tails following each other.

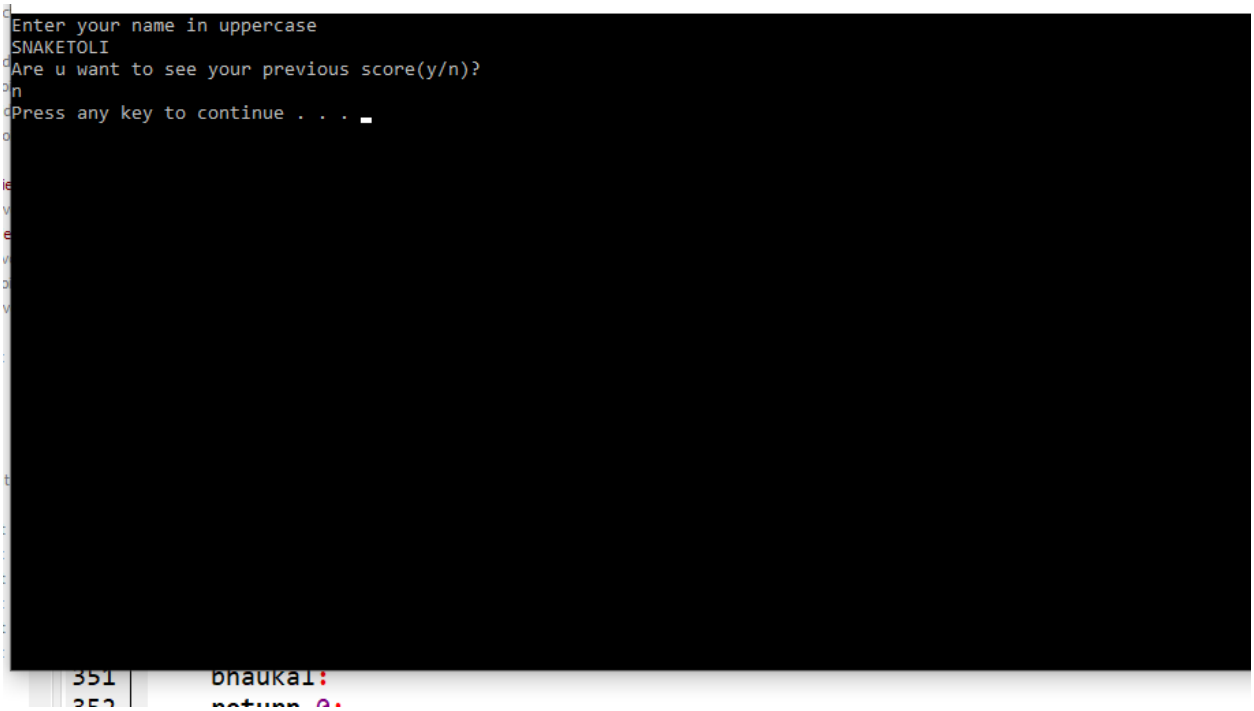
Logic():

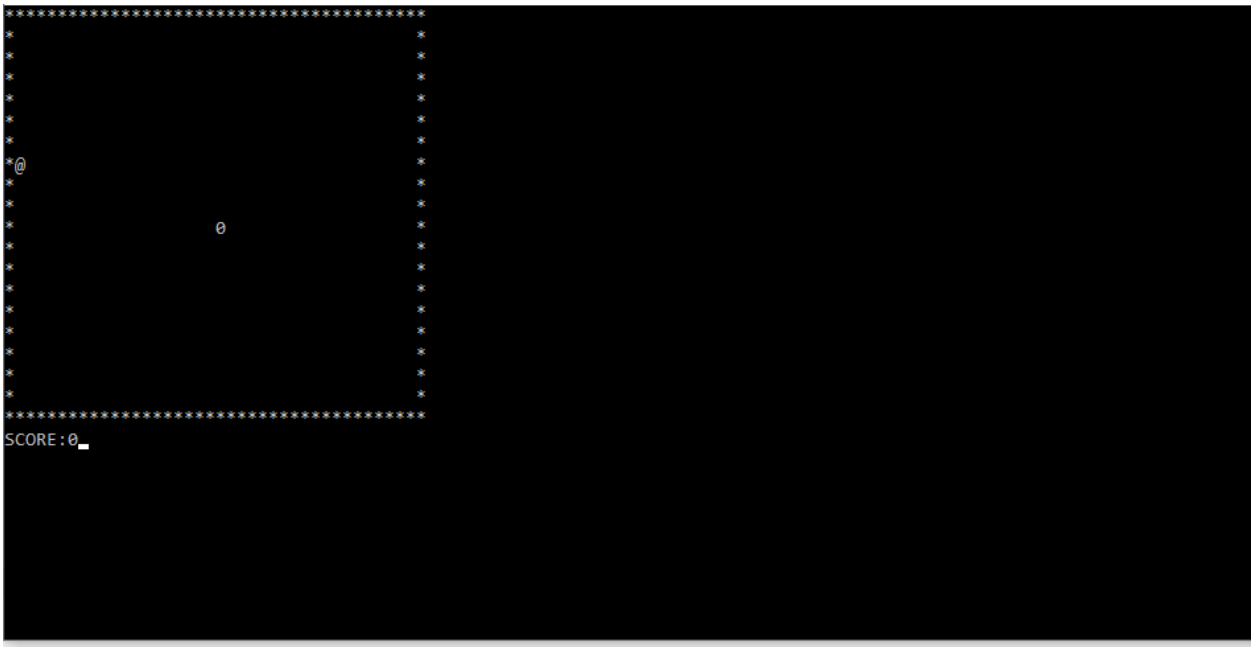
This function controls the pressing of key like, when we press certain character it excutes getting character using

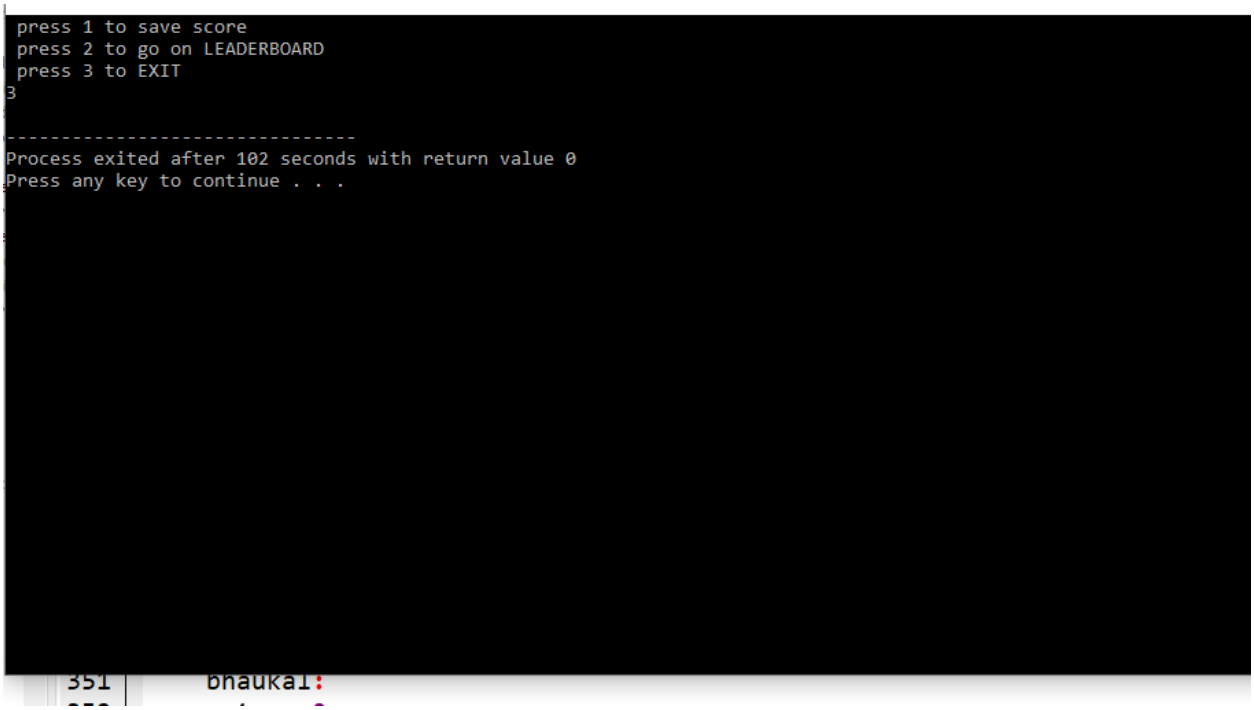
Kbhit() function.

Basically these functions are used in different forms as per required in the program. Some of them are used twice and thrice as the requirement.

Pictorial representation.







Source code

```
#include<stdio.h>
#include<stdlib.h>
#include<conio.h>
```

```
#include<string.h>
typedef struct game{
	char Name[100];
	int SCORE;
}p;
//-----------------------LEVEL1----------------------------------------------
int height=20,width=40,x,y,fruitx,fruity;
char flag;
int tailx[100],taily[100];
int prevx,prevy,prev2x,prev2y;

void fruit(){
	here:
	fruitx=rand()%40;
	if(fruitx==0||fruitx==39)
	goto here;
	there:
	fruity=rand()%20;
	if(fruity==0||fruity==19)
	goto there;
}
void snake(){
```

```
        x=width/2;
        y=height/2;
}
void setup(int piece){
        for(int i=0;i<height;i++){
                for(int j=0;j<width;j++){
                        if(i==0||i==19){
                                printf("*");
                        }
                        else if(i>0||i<19){
                                if(j==0||j==39){
                                        printf("*");
                                }
                                else if(i==y&&j==x){
                                        printf("0");
                                }
                                else if(i==fruity&&j==fruitx){
                                        printf("@");

                                }
                                else{
                                        int ch=0;
```

```
                                for(int k=1;k<=piece;k++){
                                        if(i==taily[k]&&j==tailx[k]){
                                                printf("o");
                                                ch++;
                                        }
                                }
                                if(ch==0){
                                printf(" ");}
                        }
                }


                }
                printf("\n");
        }
}
void move(int piece){
   tailx[0]=x;
        taily[0]=y;
        prevx=tailx[0];
        prevy=taily[0];
        for(int i=1;i<=piece;i++){
                prev2x=tailx[i];
```

```
            prev2y=taily[i];
            tailx[i]=prevx;
            taily[i]=prevy;
            prevx=prev2x;
            prevy=prev2y;
        }
        switch (flag){
            case 'w':
                x++;
                break;
            case 'q':
               x--;
                break;
            case 'e':
              y--;
              break;
            case  'd':
               y++;
                break;
        }
}
void logic(){
```

```
        if(kbhit()){
                flag=getch();
        }
}
//-----------------------------------------LEVEL2-----------------------------------
--------------

int height2 =20,width2=30;
int x2,y2,fruit2x,fruit2y;
int tail2x[100],tail2y[100],prev3x,prev3y,prev4x,prev4y;
int piece2=0;
void fruit2(){
        label3:
        fruit2x=rand()%30;
        if(fruit2x==0||fruit2x==29){
                goto label3;
        }
        label2:
        fruit2y=rand()%20;
        if(fruit2y==8||fruit2y==0||fruit2y==19){
                goto label2;
        }
```

```
}
void snake2(){
	x2=width2/2;
	y2=height2/2;
}
void move2(){
	tail2x[0]=x2;
	tail2y[0]=y2;
	prev3x=tail2x[0];
	prev3y=tail2y[0];
	for(int j=1;j<=piece2;j++){
		prev4x=tail2x[j];
		prev4y=tail2y[j];
		tail2x[j]=prev3x;
		tail2y[j]=prev3y;
		prev3x=prev4x;
		prev3y=prev4y;
	}
	switch(flag){
		case 'w':
			x2++;
			break;
```

```
                case 'q':
                  x2--;
                  break;
                case  'e':
                   y2--;
                   break;
                case  'd':
                   y2++;
                   break;
        }
}
void logic2(){
        if(kbhit()){
                flag=getch();
        }
}
void setup2(){
        for(int i=0;i<height2;i++){
                for(int j=0;j<width2;j++){
                        if(i==0||i==19){
                                printf("*");
                        }
```

```
else if(i==8){
        if(j==6||j==7){
                printf(" ");


        }
        else{
                printf("*");
        }
}
else if(i>0||i<19){
        if(j==0||j==29){
                printf("*");
        }
        else if(i==y2&&j==x2){
                printf("*");
        }
        else if(i==fruit2y&&j==fruit2x){
                printf("0");
        }
        else{
                int m=0;
                for(int k=1;k<=piece2;k++){
```

```
                                        printf("o",tail2x[k],tail2y[k]);
                                        m++;
                                    }
                                    if(m==0){

                                    printf(" ");}
                                }
                            }
                        }
                        printf("\n");
                }}


//---------------------------------END_SETUP-----------------------------------------
----------------
int main(){

    int k=0;
        int a=1;
        int piece=0;
        //for score recording
        p  r[100],g[100];
```

```
        FILE *f;
        int i=0;
        char name[100];
        printf("Enter your name in uppercase\n");
        if(k>0){
                getchar();
        }
        gets(name);
        f=fopen("SCORE.txt","r");
        while(fscanf(f,"%s%d",r[i].Name,&r[i].SCORE)!=EOF){
                i++;
        }
        printf("Are u want to see your previous score(y/n)?\n");
        char ansu;
        scanf("%c",&ansu);
        if(i>0){
  if(ansu=='y'||ansu=='Y'){
        for(int j=0;j<i;j++){
                if(strcmp(name,r[j].Name)==0){
                        printf("%s \n----------\nHIGH
SCORE:%d\n",r[j].Name,r[j].SCORE);
                        goto goa;
```

```
                }
        }
        printf("SORRY, NO SCORE SAVED ON THIS NAME\n");
}}
else{
    goto goa;
}
goa:
    fclose(f);
    system("pause");
    system("cls");
    int game_end=0;
    int score=0;
    snake();
    fruit();
    while(game_end!=1){
    setup(piece);
    printf("SCORE:%d",score);
    move(piece);
    logic();
    if(x==fruitx&&y==fruity){
            score=score+10;
```

```
            piece=piece+1;
            fruit();
        }
        else if(x==39||x==0||y==19||y==0){
            goto ayodha;//line 208
        }
        else if(score>30){
            goto lanka;
        }
        system("cls");}
        lanka:
         system("cls");
         printf("-----------CONRATULATIONS !!! LEVEL COMPLETED----------
-----\n");
         printf("Enter any key to go next level");
         system("pause");
         system("cls");
                    fruit2();
                    snake2();

            while(game_end==0){
            setup2();
```

```
            printf("SCORE:%d",score);
            move2();
            logic2();
            if(x2==fruit2x&&y2==fruit2y){
                  fruit2();
                  score=score+10;
                  piece2++;
            }
            else
if(x2==0||x2==29||y2==0||y2==19||(y2==8&&(x2<6||x2>7)))
            {
                  game_end++;
            }
            system("cls");}



      //
      ayodha:
      int l;
printf("THANK YOU FOR PLAYING\n");
mirjapur:
      system("cls");
```

```
printf(" press 1 to save score\n press 2 to go on LEADERBOARD\n press 3 to EXIT\n");
scanf("%d",&l);
switch(l){
        case 1:
                system("cls");
                f=fopen("SCORE.txt","w");
                if(i==0){
                    strcpy(r[0].Name,name);
                        r[0].SCORE=score;
                        i=i+1;
                        fprintf(f,"%s\t%d",r[0].Name,r[0].SCORE);
                }
                else if(i>0){
                    for(int j=0;j<i;j++){
                        if(strcmp(r[j].Name,name)==0){
                        if(r[j].SCORE<score){
                                r[j].SCORE=score;
                                 }
                                goto compex;
                                 }
                         }
```

```
                i=i+1;
            strcpy(r[i].Name,name);
             r[i].SCORE=score;

             compex:
            for(int j=0;j<i;j++){
                for(int b=0;b<i;b++){
                    if(r[b].SCORE<r[b+1].SCORE){
                        g[b]=r[b];
                        strcpy(r[b].Name,r[b+1].Name);
                        r[b].SCORE=r[b+1].SCORE;
                        r[b+1]=g[b];
                    }
                }
            }
        for(int j=0;j<i;j++){
            fprintf(f,"%s\t%d\n",r[j].Name,r[j].SCORE);
        }

        }
        printf("SCORE saved successfully\nPRESS any key to go back
menu");
```

```
            getch();
            goto mirjapur;//line 284
            break;
        case 2:
            a=1;
           system("cls");

        printf("***************LEADERBOARD*********************
**");
            printf("\n-----------------------------------------------");
            printf("\nNAME\tSCORE\tRANK\n");
            printf("-----------------------------------------------\n");
            for(int j=0; j<i;j++)       {
            printf("%s\t%d\t%d\n ",r[j].Name,r[j].SCORE,a);
            a++;// use for writing rank
            }
            system("pause");
            goto mirjapur;//line 284
            break;
        case   3:
           goto bhaukal;//line 350
            break;
```

```
}
   bhaukal:
        return 0;
}
```

……

Thank you