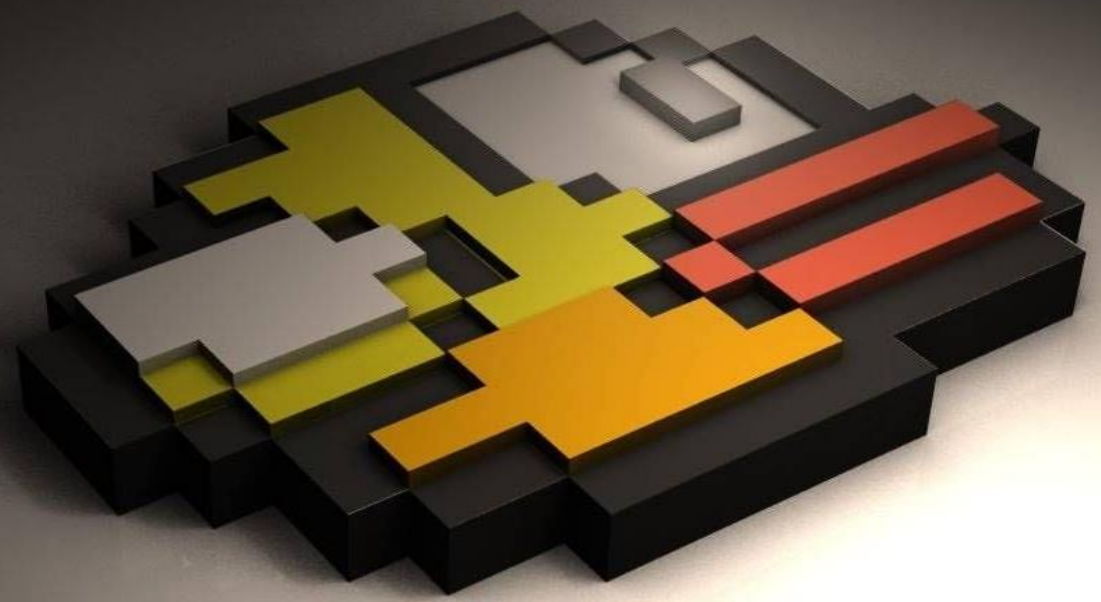
PRESENTED BY:

-PRATYUSH KUMAR

-PRADEEP KUMAR

FLAPPY BIRD GAME USING C++



ABOUT THIS GAME

Flappy Bird is an arcade-style game in which the player controls the bird FABY, which moves persistently to the right. The player is tasked with navigating FABY through pairs of pipes that have equally sized gaps placed at random heights. FABY automatically descends and only ascends when the player taps the touchscreen. Each successful pass through a pair of pipes awards the player one point. Colliding with a pipe or the ground ends the gameplay. During the GAME OVER screen, the player is awarded a bronze medal if they reached ten or more points, a silver medal from twenty points, a gold medal from thirty points, and a platinum medal from forty points

CODE

sers > kumar > OneDrive > Desktop > miniproject > flappybird.cpp >

```cpp
#include<iostream>
#include<conio.h>
#include<dos.h>
#include<stdlib.h>
#include<string.h>
#include <windows.h>
#include <time.h>

#define SCREEN_WIDTH 90
#define SCREEN_HEIGHT 26
#define WIN_WIDTH 70
#define MENU_WIDTH 20
#define GAP_SIZE 7
#define PIPE_DIF 45

using namespace std;
//      cout<<"□□□□□□□□□□";

HANDLE console = GetStdHandle(STD_OUTPUT_HANDLE);
COORD CursorPosition;

int pipePos[3];
int gapPos[3];
int pipeFlag[3];
char bird[2][6] = { '/','-','-','o','\\',' ',
                    '|','_','_','_',' ','>' };
int birdPos = 6;
int score = 0;

void gotoxy(int x, int y)
{
    CursorPosition.X = x;
    CursorPosition.Y = y;
    SetConsoleCursorPosition(console, CursorPosition);
}

void setcursor(bool visible, DWORD size)
{
    if(size == 0)
        size = 20;

    CONSOLE_CURSOR_INFO lpCursor;
    lpCursor.bVisible = visible;
    lpCursor.dwSize = size;
    SetConsoleCursorInfo(console,&lpCursor);
}

void drawBorder(){

    for(int i=0; i<SCREEN_WIDTH; i++){
        gotoxy(i,0); cout<<"□";
        gotoxy(i,SCREEN_HEIGHT); cout<<"□";
    }

    for(int i=0; i<SCREEN_HEIGHT; i++){
        gotoxy(0,i); cout<<"□";
        gotoxy(SCREEN_WIDTH,i); cout<<"□";
    }
    for(int i=0; i<SCREEN_HEIGHT; i++){
        gotoxy(WIN_WIDTH,i); cout<<"□";
    }
}
void genPipe(int ind){
    gapPos[ind] = 3 + rand()%14;
}
```

```
void drawPipe(int ind){
    if( pipeFlag[ind] == true ){
        for(int i=0; i<gapPos[ind]; i++){
            gotoxy(WIN_WIDTH-pipePos[ind],i+1); cout<<"***";
        }
        for(int i=gapPos[ind]+GAP_SIZE; i<SCREEN_HEIGHT-1; i++){
            gotoxy(WIN_WIDTH-pipePos[ind],i+1); cout<<"***";
        }
    }
}
void erasePipe(int ind){
    if( pipeFlag[ind] == true ){
        for(int i=0; i<gapPos[ind]; i++){
            gotoxy(WIN_WIDTH-pipePos[ind],i+1); cout<<"   ";
        }
        for(int i=gapPos[ind]+GAP_SIZE; i<SCREEN_HEIGHT-1; i++){
            gotoxy(WIN_WIDTH-pipePos[ind],i+1); cout<<"   ";
        }
    }
}

void drawBird(){
    for(int i=0; i<2; i++){
        for(int j=0; j<6; j++){
            gotoxy(j+2,i+birdPos); cout<<bird[i][j];
        }
    }
}
```

```
void eraseBird(){
    for(int i=0; i<2; i++){
        for(int j=0; j<6; j++){
            gotoxy(j+2,i+birdPos); cout<<" ";
        }
    }
}

int collision(){
    if( pipePos[0] >= 61  ){
        if( birdPos<gapPos[0] || birdPos >gapPos[0]+GAP_SIZE ){
//          cout<< " HIT ";
//          getch();
            return 1;
        }
    }
    return 0;
}
void debug(){
//  gotoxy(SCREEN_WIDTH + 3, 4); cout<<"Pipe Pos: "<<pipePos[0];

}
void gameover(){
    system("cls");
    cout<<endl;
    cout<<"\t\t--------------------------"<<endl;
    cout<<"\t\t-------- Game Over -------"<<endl;
    cout<<"\t\t--------------------------"<<endl<<endl;
    cout<<"\t\tPress any key to go back to menu.";
    getch();
}
```

```
void updateScore(){
    gotoxy(WIN_WIDTH + 7, 5);cout<<"Score: "<<score<<endl;
}

void instructions(){

    system("cls");
    cout<<"Instructions";
    cout<<"\n----------------";
    cout<<"\n Press spacebar to make bird fly";
    cout<<"\n\nPress any key to go back to menu";
    getch();
}

void play(){

    birdPos = 6;
    score = 0;
    pipeFlag[0] = 1;
    pipeFlag[1] = 0;
    pipePos[0] = pipePos[1] = 4;

    system("cls");
    drawBorder();
    genPipe(0);
    updateScore();
    gotoxy(WIN_WIDTH + 5, 2);cout<<"FLAPPY BIRD";
    gotoxy(WIN_WIDTH + 6, 4);cout<<"----------";
    gotoxy(WIN_WIDTH + 6, 6);cout<<"----------";
    gotoxy(WIN_WIDTH + 7, 12);cout<<"Control ";
    gotoxy(WIN_WIDTH + 7, 13);cout<<"-------- ";
    gotoxy(WIN_WIDTH + 2, 14);cout<<" Spacebar = jump";

    gotoxy(10, 5);cout<<"Press any key to start";
    getch();
    gotoxy(10, 5);cout<<"                      ";

    while(1){

        if(kbhit()){
            char ch = getch();
            if(ch==32){
                if( birdPos > 3 )
                    birdPos-=3;
            }
            if(ch==27){
                break;
            }
        }

        drawBird();
        drawPipe(0);
        drawPipe(1);
        debug();
        if( collision() == 1 ){
            gameover();
            return;
        }
```

```
    drawBird();
    drawPipe(0);
    drawPipe(1);
    debug();
    if( collision() == 1 ){
        gameover();
        return;
    }
    Sleep(100);
    eraseBird();
    erasePipe(0);
    erasePipe(1);
    birdPos += 1;

    if( birdPos > SCREEN_HEIGHT - 2 ){
        gameover();
        return;
    }

    if( pipeFlag[0] == 1 )
        pipePos[0] += 2;

    if( pipeFlag[1] == 1 )
        pipePos[1] += 2;

    if( pipePos[0] >= 40 && pipePos[0] < 42 ){
        pipeFlag[1] = 1;
        pipePos[1] = 4;
        genPipe(1);
    }
    if( pipePos[0] > 68 ){
        score++;
                score++;
                updateScore();
                pipeFlag[1] = 0;
                pipePos[0] = pipePos[1];
                gapPos[0] = gapPos[1];
            }

        }

}

int main()
{
    setcursor(0,0);
    srand( (unsigned)time(NULL));

//  play();

    do{
        system("cls");
        gotoxy(10,5); cout<<" ------------------------- ";
        gotoxy(10,6); cout<<" |      Flappy Bird       | ";
        gotoxy(10,7); cout<<" -------------------------";
        gotoxy(10,9); cout<<"1. Start Game";
        gotoxy(10,10); cout<<"2. Instructions";
        gotoxy(10,11); cout<<"3. Quit";
        gotoxy(10,13); cout<<"Select option: ";
        char op = getche();

        if( op=='1') play();
        else if( op=='2') instructions();
        else if( op=='3') exit(0);
```

```
    do{
        system("cls");
        gotoxy(10,5); cout<<" -------------------------- ";
        gotoxy(10,6); cout<<" |      Flappy Bird       | ";
        gotoxy(10,7); cout<<" --------------------------";
        gotoxy(10,9); cout<<"1. Start Game";
        gotoxy(10,10); cout<<"2. Instructions";
        gotoxy(10,11); cout<<"3. Quit";
        gotoxy(10,13); cout<<"Select option: ";
        char op = getche();

        if( op=='1') play();
        else if( op=='2') instructions();
        else if( op=='3') exit(0);

    }while(1);
    return 0;
}
```

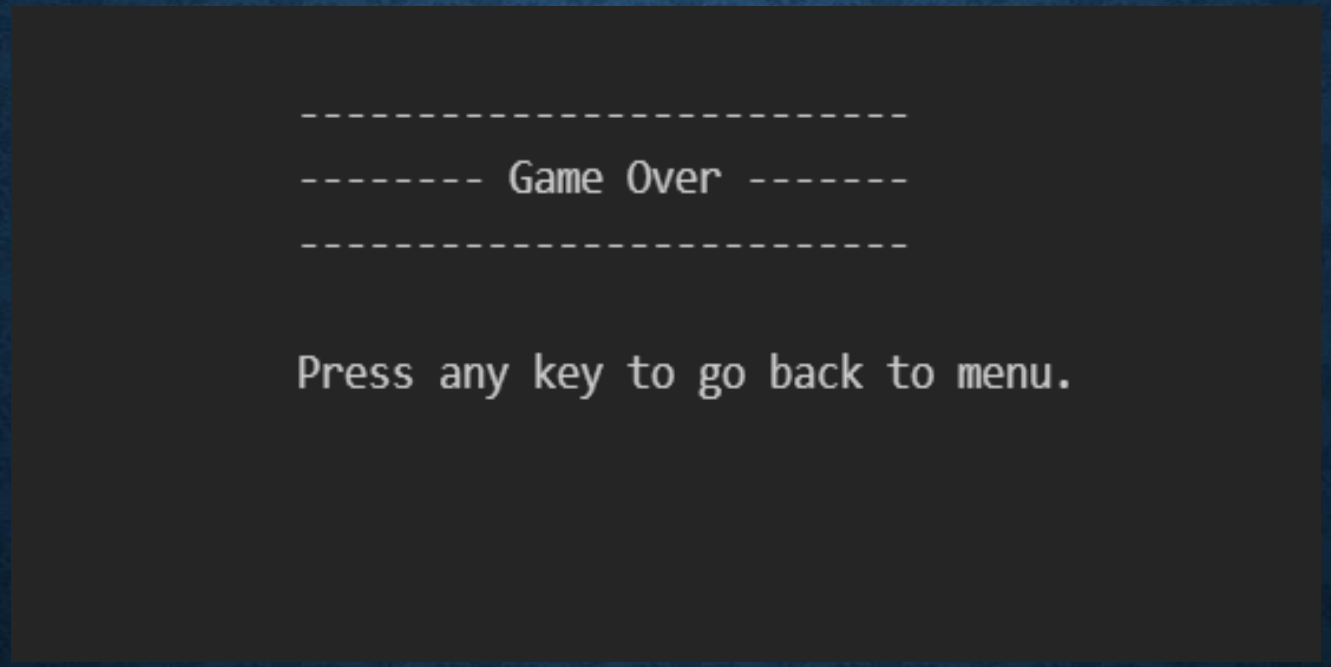
OUTPUT

```
 --------------------------
 |      Flappy Bird       |
 --------------------------

1. Start Game
2. Instructions
3. Quit

Select option:
```

Instructions

Press spacebar to make bird fly

Press any key to go back to menu


--------- Game Over --------
Press any key to go back to menu.

BIBLOGRAOHY:

VISUAL STDIO CODE

GOOGLE.COM

GRAPHICS

C++ LANGUAGE

THANK YOU

****