ACKNOWLEDGEMENTS

**Firstly , I would like to thank our beloved lord Bhagawan because of whose grace this project is a reality today.**

**Next , I would like to thank my computer science teacher , SRI VENKATESWAR PRUSTY , who has provided me constant guidance throughout my project.**

**I would also like to thank our principal , SRI SHIVA RAMKRISHNAIAH , for giving me this chance and also the time to complete this project.**

**Also , I would like to thank my classmates Abhishek, Ajit, Durga Prasad, Nishant, Rishi Barai, Nandan, Veerendra, Vishnu Vardhan, Vishwanath, Durgesh and Shikhar for their constant support , encouragement and also their valuable inputs.**

**THEORY**

**This project is very similar to the very common and famous game PICTURE PUZZLE. The only difference is that in the original game, the player moves the piece of picture directly above, below, to the left or to the right of the white block using the respective direction keys whereas in this game, the player moves the white block itself using the direction keys. The highest scorer is the one who completes the puzzle in the least number of moves. There are three different high score records for the three different levels.**

**The game has three levels. The first level is a 3 x 3 puzzle, the second a 4 x 4 and the third, a 5 x 5 puzzle. The game is a bit logical but it also provides a lot of entertainment. The first level is an easy one but as the levels get going the game gets tougher. I hope that everyone who plays this game enjoys it a lot.**

**AIM**

**To simulate the game**

**‘PICTURE PUZZLE’**

**Using DEV C++.**

**INTRODUCTION**

**Games have been the common interest of people of all age groups for a long time now. Games provide relaxation to peoples’ mind in between their usually boring and hectic schedules. Games provide entertainment through challenges and that is another reason why a lot of people love to play games. There are different types of games like logical, entertainment and so on. Logical games require the use of the mind to play the game whereas entertainment games are made just for the sake of entertainment. Logical games help to improve your brain whereas entertainment games help to relax your mind.**

**This project is the simulation of a mainly logical game but it also provides entertainment as there is a competition to become one of the top five high scorers by completing the puzzle in the least number of moves.**

**Program Design**

**Our computer does a lot of work for us in accordance with a set of rules or what is called a program given to it. Our computer can be made to work more efficiently if the programs are made easier to execute. That is the reason why this program too makes use of certain classes and functions which make it easier for the computer.**

**CODE**

**#include <allegro.h>**

**#include<string.h>**

**#include<alfont.h>**

**#include<dos.h>**

**void init();**

**void deinit();**

**////////////////////////////// GLOBAL VARIABLES //////////////////////////////**

**BITMAP \*bit[6],\*blank,\*bmp[25],\*bkg,\*map[6];**

**PALETTE palette;**

**int k[25];**

**int a,x,l=2,image,t,level,moves;**

**PACKFILE \*file;**

**ALFONT\_FONT \*fnt,\*fnt1,\*fnt2;**

**////////////////////////////// HIGH SCORE STRUCTURE //////////////////////////////**

**struct hiscore**

**{**

**int score;**

**char name[25];**

**}players[5];**

**////////////////////////////// FUNCTION TO CHANGE HIGH SCORE //////////////////////////////**

**void change\_highscore(int k,int o)**

**{**

**int na=0;**

**char ch;**

**bkg=load\_bitmap("yosemite-4.bmp",palette);**

**for(int j=4;j>k;j--)**

**{**

**players[j].score=players[j-1].score;**

**strcpy(players[j].name,players[j-1].name);**

**}**

**blit(bkg,screen,0,0,0,0,bkg->w,bkg->h);**

**do**

**{**

**alfont\_textout\_ex(screen,fnt1,"NEW HIGH SCORE",50,50,makecol(255,0,0),-1);**

**alfont\_textout\_ex(screen,fnt1,"ENTER NAME",125,85,makecol(255,0,0),-1);**

**ch=readkey();**

**if(!key[KEY\_ENTER] && !key[KEY\_ENTER\_PAD] && !key[KEY\_BACKSPACE])**

**{**

**players[k].name[na]=ch;**

**players[k].name[na+1]='\0';**

**alfont\_textprintf\_ex(screen,fnt1,200,150,makecol(0,0,0),-1,"%s",players[k].name);**

**na++;**

**}**

**if(key[KEY\_BACKSPACE] && na>0)**

**{**

**players[k].name[--na]='\0';**

**clear(screen);**

**blit(bkg,screen,0,0,0,0,bkg->w,bkg->h);**

**alfont\_textprintf\_ex(screen,fnt1,200,150,makecol(0,0,0),-1,"%s",players[k].name);**

**}**

**}while(!key[KEY\_ENTER] && !key[KEY\_ENTER\_PAD]);**

**players[k].score=moves;**

**switch(o)**

**{**

**case 1:file=pack\_fopen("scores\\scores1.txt","w");break;**

**case 2:file=pack\_fopen("scores\\scores2.txt","w");break;**

**case 3:file=pack\_fopen("scores\\scores3.txt","w");break;**

**}**

**for(int i=0;i<5;i++)**

**pack\_fwrite(&players[i],sizeof(hiscore),file);**

**pack\_fclose(file);**

**}**

**////////////////////////////// FUNCTION FOR DISPLAYING HIGH SCORE //////////////////////////////**

**void display\_highscore(int u)**

**{**

**switch(u)**

**{**

**case 1:file=pack\_fopen("scores\\scores1.txt","r"); break;**

**case 2:file=pack\_fopen("scores\\scores2.txt","r"); break;**

**case 3:file=pack\_fopen("scores\\scores3.txt","r"); break;**

**}**

**pack\_fseek(file,0);**

**clear\_bitmap(screen);**

**bkg=load\_bitmap("yosemite-5.bmp",palette);**

**blit(bkg,screen,0,0,0,0,bkg->w,bkg->h);**

**for(int i=0;i<5;i++)**

**pack\_fread(&players[i],sizeof(hiscore),file);**

**pack\_fclose(file);**

**alfont\_textprintf\_ex(screen,fnt2,150,20,makecol(0,255,0),-1,"LEVEL %d",u);**

**alfont\_textprintf\_ex(screen,fnt1,200,150,makecol(255,0,0),-1,"NAME");**

**alfont\_textprintf\_ex(screen,fnt1,325,150,makecol(255,0,0),-1,"MOVES");**

**for(int i=0,xco=135,yco=200;i<5;yco+=50,i++)**

**{**

**alfont\_textprintf\_ex(screen,fnt,xco+55,yco,makecol(255,255,255),-1,"%s",players[i].name);**

**alfont\_textprintf\_ex(screen,fnt,xco+220,yco,makecol(255,255,255),-1,"%d",players[i].score);**

**}**

**}**

**void after\_select(int start)**

**{**

**int width=160,height=210;**

**BITMAP \*temp=create\_bitmap(SCREEN\_W,SCREEN\_H);**

**for(int i=0,x1=40,x2=200,y1=20,y2=230;i<6;i++)**

**{**

**if(i!=image)**

**{**

**rectfill(screen,x1,y1,x2,y2,makecol(0,0,0));**

**rest(500);**

**}**

**x1+=200;x2+=200;**

**if(i==2)**

**{**

**x1=40;x2=200; y1=250;y2=460;**

**}**

**}**

**clear(screen);**

**while(start<=360)**

**{**

**clear\_bitmap(temp);**

**blit(map[image],temp,0,0,start,50,map[image]->w,map[image]->h);**

**blit(temp,screen,0,0,0,0,SCREEN\_W,SCREEN\_H);**

**start++;**

**rest(10);**

**}**

**while(width<270 || height<375)**

**{**

**stretch\_blit(map[image],screen,0,0,map[image]->w,map[image]->h,360,50,width,height);**

**if(width<270) width++;**

**if(height<375) height++;**

**rest(10);**

**}**

**}**

**////////////////////////////// CLASS WITH ALL THE ATTRIBUTES OF THE PUZZLES //////////////////////////////**

**class game**

**{**

**public:**

**void shuffle(int z)**

**{**

**for(int h=0,v;h<150;h++)**

**{**

**x=(rand()%4);**

**v=x;**

**if(x!=(v+1) && x!=(v-1))**

**if(x==0 && l>z-1 && k[l-z]<(z\*z))**

**{**

**k[l]=(k[l]\*k[l-z])/(k[l-z]=k[l]);**

**l-=z;**

**}**

**else if(x==1 && l<z\*(z-1) && k[l+z]<(z\*z))**

**{**

**k[l]=(k[l]\*k[l+z])/(k[l+z]=k[l]);**

**l+=z;**

**}**

**else if(x==2 && (l%z)!=0 && k[l-1]<(z\*z))**

**{**

**k[l]=(k[l]\*k[l-1])/(k[l-1]=k[l]);**

**l-=1;**

**}**

**else if(x==3 && ((l+1)%z)!=0 && k[l+1]<(z\*z))**

**{**

**k[l]=(k[l]\*k[l+1])/(k[l+1]=k[l]);**

**l+=1;**

**}**

**}**

**}**

**void display()**

**{**

**for(int u=0,c=40,d=20;u<6;u++)**

**if(c<640)**

**{**

**blit(map[u],screen,0,0,c,d,map[u]->w,map[u]->h);**

**c+=200;**

**}**

**else**

**{**

**c=40;**

**d+=230;**

**blit(map[u],screen,0,0,c,d,map[u]->w,map[u]->h);**

**c+=200;**

**}**

**}**

**int select()**

**{**

**while(1)**

**{**

**if(mouse\_x>40 && mouse\_x<200 && mouse\_y>20 && mouse\_y<230)**

**{**

**if(mouse\_b==1)**

**{**

**image=0;**

**return 40;break;**

**}**

**}**

**else if(mouse\_x>240 && mouse\_x<400 && mouse\_y>20 && mouse\_y<230)**

**{**

**if(mouse\_b==1)**

**{**

**image=1;**

**return 240;break;**

**}**

**}**

**else if(mouse\_x>440 && mouse\_x<600 && mouse\_y>20 && mouse\_y<230)**

**{**

**if(mouse\_b==1)**

**{**

**image=2;**

**return 440;break;**

**}**

**}**

**else if(mouse\_x>40 && mouse\_x<200 && mouse\_y>250 && mouse\_y<460)**

**{**

**if(mouse\_b==1)**

**{**

**image=3;**

**return 40;break;**

**}**

**}**

**else if(mouse\_x>240 &&mouse\_x<400 && mouse\_y>250 && mouse\_y<460)**

**{**

**if(mouse\_b==1)**

**{**

**image=4;**

**return 240;break;**

**}**

**}**

**else if(mouse\_x>440 && mouse\_x<600 && mouse\_y>250 && mouse\_y<460)**

**{**

**if(mouse\_b==1)**

**{**

**image=5;**

**return 440;break;**

**}**

**}**

**else if(key[KEY\_ESC])**

**{ return 0;break; }**

**}**

**}**

**void load\_real()**

**{**

**bit[0]=load\_bitmap("bmp\\school logo.bmp",palette);**

**bit[1]=load\_bitmap("bmp\\pic.bmp",palette);**

**bit[2]=load\_bitmap("bmp\\jones.bmp",palette);**

**bit[3]=load\_bitmap("bmp\\crime1.bmp",palette);**

**bit[4]=load\_bitmap("bmp\\crime2.bmp",palette);**

**bit[5]=load\_bitmap("bmp\\crime3.bmp",palette);**

**}**

**void load()**

**{**

**map[0]=load\_bitmap("bmp\\map1.bmp",palette);**

**map[1]=load\_bitmap("bmp\\map2.bmp",palette);**

**map[2]=load\_bitmap("bmp\\map3.bmp",palette);**

**map[3]=load\_bitmap("bmp\\map4.bmp",palette);**

**map[4]=load\_bitmap("bmp\\map5.bmp",palette);**

**map[5]=load\_bitmap("bmp\\map6.bmp",palette);**

**}**

**void cut(int y)**

**{**

**blank = load\_bitmap("bmp\\Blank.bmp",palette);**

**int w,v;**

**w=270/y; v=375/y;**

**for(int a=0,e=0,r=0;a<(y\*y);a++)**

**{**

**bmp[a]=create\_bitmap(w,v);**

**if(a!=y-1)**

**blit(bit[image],bmp[a],e,r,0,0,w,v);**

**else**

**blit(blank,bmp[a],0,0,0,0,w,v);**

**if((a+1)%y==0)**

**{ e=0;r+=v; }**

**else**

**e+=w;**

**}**

**}**

**int check(int a)**

**{**

**int i;**

**for(i=0;i<(a\*a);i++)**

**if(i!=k[i])**

**break;**

**return i;**

**}**

**void play(int a)**

**{**

**int m,n;**

**m=270/a; n=375/a;**

**for(int q=0,r=50,e=50;q<(a\*a);q++)**

**{**

**blit(bmp[k[q]],screen,0,0,r,e,bmp[k[q]]->w,bmp[k[q]]->h);**

**if((q+1)%a==0)**

**{ r=50; e+=n; }**

**else**

**r+=m;**

**}**

**readkey();**

**if(key[KEY\_UP])**

**{ if(l>(a-1) && k[l-a]<(a\*a))**

**{**

**k[l]=(k[l]\*k[l-a])/(k[l-a]=k[l]);**

**l-=a;**

**moves++;**

**}**

**}**

**else if(key[KEY\_DOWN])**

**{ if(l<(a\*(a-1)) && k[l+a]<(a\*a))**

**{**

**k[l]=(k[l]\*k[l+a])/(k[l+a]=k[l]);**

**l+=a;**

**moves++;**

**}**

**}**

**else if(key[KEY\_LEFT])**

**{ if(l%a!=0 && k[l-1]<(a\*a))**

**{**

**k[l]=(k[l]\*k[l-1])/(k[l-1]=k[l]);**

**l-=1;**

**moves++;**

**}**

**}**

**else if(key[KEY\_RIGHT])**

**{ if((l+1)%a!=0 && k[l+1]<(a\*a))**

**{**

**k[l]=(k[l]\*k[l+1])/(k[l+1]=k[l]);**

**l+=1;**

**moves++;**

**}**

**}**

**else if(key[KEY\_R])**

**{**

**if(check(a)!=(a\*a))**

**moves=0;**

**for(int g=0;g<25;g++)**

**k[g]=g;**

**int q;**

**do**

**{**

**q=0;**

**shuffle(a);**

**for(int r=0;r<(a\*a);r++)**

**if(r==k[r])**

**{**

**q=1;**

**break;**

**}**

**}while(q==1);**

**assign(a);**

**}**

**}**

**void assign(int u)**

**{**

**for(a=0;a<(u\*u);a++)**

**if(k[a]==(u-1))**

**{**

**l=a;**

**break;**

**}**

**}**

**}puzzle1,puzzle2,puzzle3;**

**////////////////////////////// START OF MAIN //////////////////////////////**

**int main() {**

**init();**

**alfont\_init();**

**srand(time(NULL));**

**int y,q,i=200,chek;**

**BITMAP \*intro,\*levels;**

**intro=load\_bitmap("bmp\\page1.bmp",palette);**

**levels=load\_bitmap("bmp\\levels.bmp",palette);**

**fnt=alfont\_load\_font("C:\\WINDOWS\\Fonts\\ALGER.ttf");**

**fnt1=alfont\_load\_font("C:\\WINDOWS\\Fonts\\ALGER.ttf");**

**fnt2=alfont\_load\_font("C:\\WINDOWS\\Fonts\\ALGER.ttf");**

**alfont\_set\_font\_size(fnt,20);**

**alfont\_set\_font\_size(fnt1,30);**

**alfont\_set\_font\_size(fnt2,100);**

**do**

**{**

**level=1; y=0;**

**for(int i=0;i<25;i++)**

**k[i]=i;**

**////////////////////////////// MAIN MENU //////////////////////////////**

**while(1)**

**{**

**blit(intro,screen,0,0,0,0,640,480);**

**readkey();**

**if(key[KEY\_P])**

**{**

**clear(screen);**

**break;**

**}**

**else if(key[KEY\_L])**

**{**

**clear(screen);**

**blit(levels,screen,0,0,0,0,640,480);**

**while(1)**

**{**

**readkey();**

**if(key[KEY\_1] || key[KEY\_1\_PAD])**

**{ level=1;break; }**

**else if(key[KEY\_2] || key[KEY\_2\_PAD])**

**{ level=2;break; }**

**else if(key[KEY\_3] || key[KEY\_3\_PAD])**

**{ level=3;break; }**

**}**

**break;**

**}**

**else if(key[KEY\_H])**

**{**

**clear(screen);**

**blit(levels,screen,0,0,0,0,640,480);**

**readkey();**

**if(key[KEY\_1] || key[KEY\_1\_PAD])**

**display\_highscore(1);**

**else if(key[KEY\_2] || key[KEY\_2\_PAD])**

**display\_highscore(2);**

**else if(key[KEY\_3] || key[KEY\_3\_PAD])**

**display\_highscore(3);**

**readkey();**

**}**

**else if(key[KEY\_E])**

**exit(0);**

**}**

**////////////////////////////// LEVEL 1 //////////////////////////////**

**if(level==1)**

**{**

**clear(screen);**

**int q;**

**do**

**{**

**q=0;**

**puzzle1.shuffle(3);**

**for(int r=0;r<9;r++)**

**if(r==k[r])**

**{**

**q=1;**

**break;**

**}**

**}while(q==1);**

**puzzle1.load\_real();**

**puzzle1.load();**

**puzzle1.display();**

**chek=puzzle1.select();**

**if(chek==0)**

**{ chek=1; goto end; }**

**after\_select(chek);**

**puzzle1.cut(3);**

**blit(bit[image],screen,0,0,360,50,bit[image]->w,bit[image]->h);**

**blit(bmp[2],screen,0,0,540,50,bmp[2]->w,bmp[2]->h);**

**puzzle1.assign(3);**

**moves=0;**

**while (!key[KEY\_ESC])**

**{**

**textprintf\_ex(screen, font,300,20 , makecol(255, 100, 200),makecol(0,0,0), "MOVES: %d", moves);**

**puzzle1.play(3);**

**if(y==9)**

**break;**

**y=puzzle1.check(3);**

**}**

**file=pack\_fopen("scores\\scores1.txt","r");**

**pack\_fseek(file,0);**

**for(int i=0;i<5;i++)**

**pack\_fread(&players[i],sizeof(hiscore),file);**

**pack\_fclose(file);**

**for(int i=0;i<5;i++)**

**if(moves<players[i].score && y==9)**

**{**

**clear(screen);**

**change\_highscore(i,1);**

**display\_highscore(1);**

**readkey();**

**break;**

**}**

**}**

**////////////////////////////// LEVEL 2 //////////////////////////////**

**else if(level==2)**

**{**

**clear(screen);**

**int q;**

**do**

**{**

**q=0;**

**puzzle2.shuffle(4);**

**for(int r=0;r<16;r++)**

**if(r==k[r])**

**{**

**q=1;**

**break;**

**}**

**}while(q==1);**

**puzzle2.load\_real();**

**puzzle2.load();**

**puzzle2.display();**

**chek=puzzle2.select();**

**if(chek==0)**

**{ chek=1; goto end; }**

**after\_select(chek);**

**puzzle2.cut(4);**

**blit(bit[image],screen,0,0,360,50,bit[image]->w,bit[image]->h);**

**blit(bmp[3],screen,0,0,563,50,bmp[3]->w,bmp[3]->h);**

**puzzle2.assign(4);**

**moves=0;**

**while(!key[KEY\_ESC])**

**{**

**textprintf\_ex(screen, font, 300, 20, makecol(255, 100,200),makecol(0,0,0), "MOVES: %d",moves);**

**puzzle2.play(4);**

**if(y==16)**

**break;**

**y=puzzle2.check(4);**

**}**

**file=pack\_fopen("scores\\scores2.txt","r");**

**pack\_fseek(file,0);**

**for(int i=0;i<5;i++)**

**pack\_fread(&players[i],sizeof(hiscore),file);**

**pack\_fclose(file);**

**for(int i=0;i<5;i++)**

**if(moves<players[i].score && y==16)**

**{**

**clear(screen);**

**change\_highscore(i,2);**

**display\_highscore(2);**

**readkey();**

**break;**

**}**

**}**

**////////////////////////////// LEVEL 3 //////////////////////////////**

**else if(level==3)**

**{**

**clear(screen);**

**int q;**

**do**

**{**

**q=0;**

**puzzle3.shuffle(5);**

**for(int r=0;r<25;r++)**

**if(r==k[r])**

**{**

**q=1;**

**break;**

**}**

**}while(q==1);**

**puzzle3.load\_real();**

**puzzle3.load();**

**puzzle3.display();**

**chek=puzzle3.select();**

**if(chek==0)**

**{ chek=1; goto end; }**

**after\_select(chek);**

**puzzle3.cut(5);**

**blit(bit[image],screen,0,0,360,50,bit[image]->w,bit[image]->h);**

**blit(bmp[4],screen,0,0,576,50,bmp[4]->w,bmp[4]->h);**

**puzzle3.assign(5);**

**moves=0;**

**while(!key[KEY\_ESC])**

**{**

**textprintf\_ex(screen, font, 300, 20, makecol(255, 100,200),makecol(0,0,0), "MOVES: %d",moves);**

**puzzle3.play(5);**

**if(y==25)**

**break;**

**y=puzzle3.check(5);**

**}**

**file=pack\_fopen("scores\\scores3.txt","r");**

**pack\_fseek(file,0);**

**for(int i=0;i<5;i++)**

**pack\_fread(&players[i],sizeof(hiscore),file);**

**pack\_fclose(file);**

**for(int i=0;i<5;i++)**

**if(moves<players[i].score && y==25)**

**{**

**clear(screen);**

**change\_highscore(i,3);**

**display\_highscore(3);**

**readkey();**

**break;**

**}**

**}**

**end: ;**

**}while(1);**

**deinit();**

**return 0;**

**}**

**END\_OF\_MAIN()**

**void init() {**

**int depth, res;**

**allegro\_init();**

**depth = desktop\_color\_depth();**

**if (depth == 0) depth = 32;**

**set\_color\_depth(depth);**

**res = set\_gfx\_mode(GFX\_AUTODETECT\_WINDOWED, 640, 480, 0, 0);**

**if (res != 0) {**

**allegro\_message(allegro\_error);**

**exit(-1);**

**}**

**install\_timer();**

**install\_keyboard();**

**install\_mouse();**

**show\_mouse(screen);**

**/\* add other initializations here \*/**

**}**

**void deinit() {**

**clear\_keybuf();**

**/\* add other deinitializations here \*/**

**}**