4/15/2018

HIT THE SPOT

Computer Graphics minor project

UNDER THE GUIDANCE OF

→DR. SOMNATH DEY

TEAM MEMBERS

→E. RANJITH KUMAR 1500001011

→THOTA SRI RANGA VINEEL 150001036

Computer Science and Engineering

IIT INDORE

Introduction:

In this minor project, we have developed an archery game.

Game Description:

In this game user can set the view according to his preference using arrow keys. And also he/she can set the speed and angle of the arrow to shoot. After that the score will be updated according the target board hit. When arrow count becomes zero the game will end.

Functionality description:

Functionality Name	Description
GLuint LoadBMP(const char *fileName)	This function takes the location of bmp image file and returns the texture id so that we can use that id whenever we want to map the texture to an object.
void initTexture()	In this function, we first loaded the sky and ground textures using above mentioned function and stored them in the variables sky and grass respectively.
void draw_circle(float rad,float r,float g,float b)	This function takes radius and color (red, green and blue) as inputs and draw the circle on the screen.
void draw_angle_board()	This function was developed to draw the angular board. At first a semicircle was drawn and then a line with one end at the center of the circle and other end on the inner radius of the semicircle was drawn. The second end point changes with respect to time, so that it looks like line is changing the direction from 170 to 10 degrees.
void speed_bar()	A speed bar will be drawn using this function. When the user presses ‘a’ the height of the bar will be varied according to that.

void Arrow()	This function was developed to draw an arrow. And when the user shoots the arrow will be redrawn in each frame with the angle and speed chosen by the user. Whenever the arrow front point reaches the plane where target board is situated the score will be upgraded as specified in this function and then starting state values will be initialized back by decreasing arrow count.
void NormalKey(GLubyte key, GLint x, GLint y)	This was used to handle with the keyboard keys and to do the action according to that.
void SpecialKey(int key, int xx, int yy)	This was used to handle the keyboard special keys (up, down, right, left) and do the action according to that.
void DrawTextXY(double x,double y,double z,double scale)	This function was developed to draw the text with the specified location and size.
void Draw_sky()	This function was used to create the sky with the sky texture.
void Draw_ground()	This function was used to create the ground with the grass texture.
void Score()	This function was developed to show the updated score on screen.
void Display(void)	This function was developed to handle the changes come in the game window and redraw according to that.

void menu1()	This function was developed to handle the starting menu.
void menu2()	This function was developed to handle the help menu.
void menu3()	This function was developed to handle the game over menu.
void Dis()	This function was called in the glutDisplayFunc to handle the changes by using above mentioned functions.
void Reshape(int x, int y)	This function handles with the window resizes.

Manual to game:

Start Menu:

In this starting menu there are three options

1)Play

2)Help

3)Exit

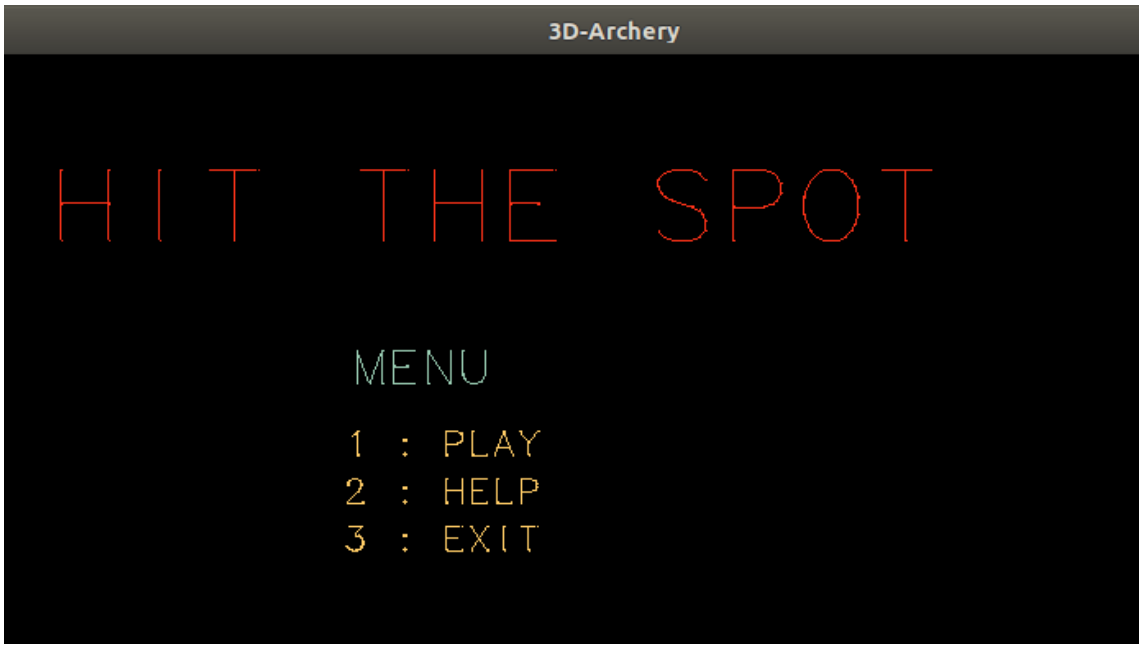
Player can choose any one of this options by entering corresponding number.

→If player chooses 'Play', he/she will be directed to main game window.

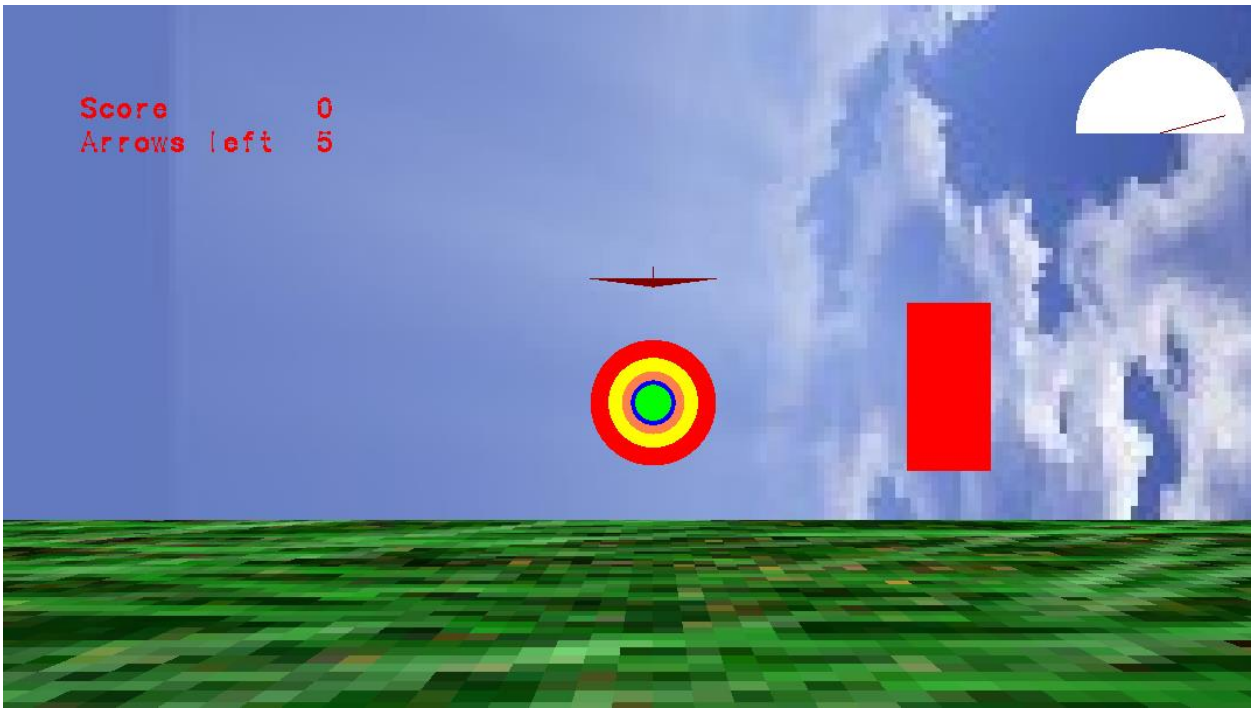
→If player chooses 'Help', he/she will be directed to help menu.

→If player chooses ‘Exit’, the game will be halted.

Main Game Window:

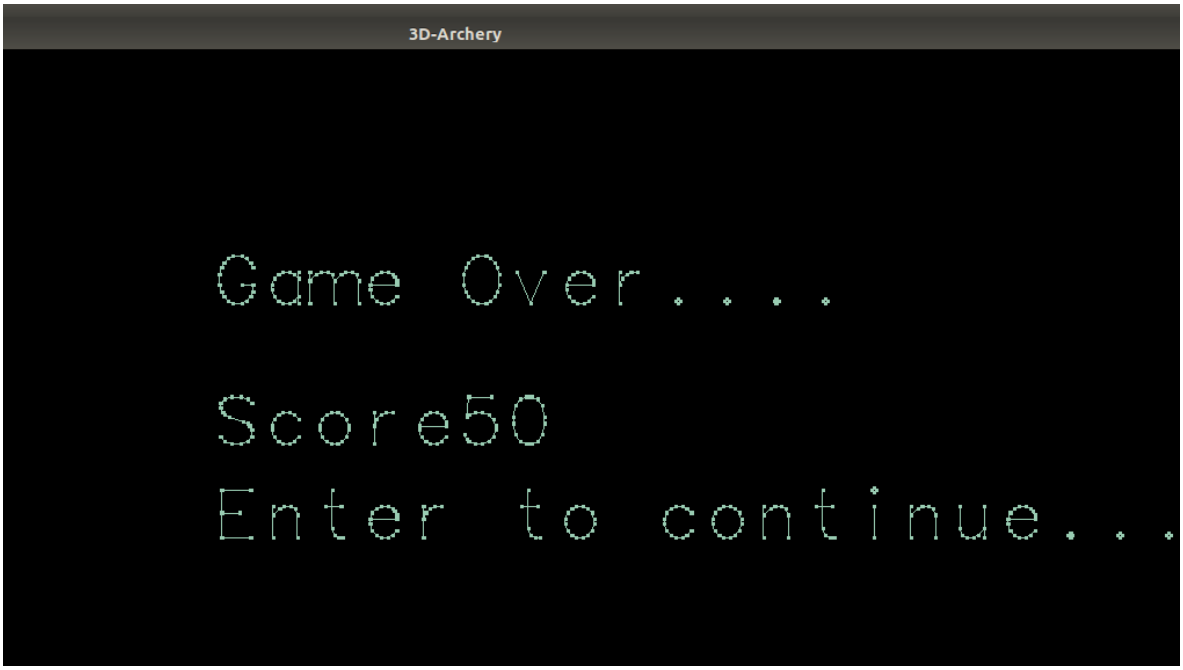


In this game, the player first can fix the position according to his/her preference by clicking right/left/top/bottom keyboard key. Then he/she has to set the speed of the arrow by pressing ‘a’ on keyboard. After that the player can shoot the arrow at the angle he/she prefers by looking at the angular board by pressing ‘Enter’ key. According to the distance from the center of the target board and the point of contact of arrow with the board, the score will be updated. And also the arrow count will be reduced by one.



Whenever the arrow count reaches zero, the game over menu will popup.

Game Over Menu:



Help Menu:

Player can press ‘Enter’ key to continue with the game.

Conclusion:

With this minor project we have learned, drawing the scenes using opengl graphics, also handling the user interactive graphics.