

ARMY CAMP VIEW

Course: Computer Graphics.
Instructor: MD Kishor Morol.
Section: H.
Group: 4

Group members:

Name	ID
Aseya Khanom	17-35394-3
Bushra Amana	18-36568-1
Shafin Talukder	18-36588-1
Arshad Hassan Aywon	18-36601-1
Nayma Islam	18-36607-1

Problem statement:

The features that have been implemented is downscaled below:

- Sun and Moon
- Sounds.
- Tent.
- Tree.
- Flag.
- Road.
- Windmill.
- Clouds.
- Helicopter.
- Helipad
- Hill view.
- Army tank.
- Day mood.
- Night mood.
- Watchtower.
- Fireworks.
- Sun in “Day” mood.
- Moon in “Night” mood.
- Season feature: Rain.
- Interaction: Bird moving.
- Interaction: Cloud moving.
- Interaction: Helicopter moving.
- Interaction: Vehicle on road moving.

The functions that have been implemented is downscaled below:

1. `glClearColor`: specify clear values for the color buffers.
2. `glPointSize`: specify the diameter of rasterized points.
3. `gluOrtho2D`: define a 2D orthographic projection matrix
4. `glVertex2i`: specify a vertex
5. `glPushMatrix`: push the current matrix stack
6. `glTranslatef`: add the current matrix by a translation matrix
7. `glColor3ub`: sets the current color
8. `glVertex2f`: Specifies a vertex
9. `glPopMatrix`: pop the current matrix stack
10. `glScalef`: multiply the current matrix by a general scaling matrix.
11. `glutTimerFunc`: registers a timer callback to be triggered in a specified number of milliseconds.
12. `glutKeyboardFunc`: sets the keyboard callback for the current window.
13. `glutInit`: A pointer to the program's unmodified `argc` variable from main. Upon return, the value pointed to by `argv` will be updated, because `glutInit` extracts any command line options intended for the GLUT library
14. `glutInitWindowSize`: set the initial window position and size respectively

Methodology:

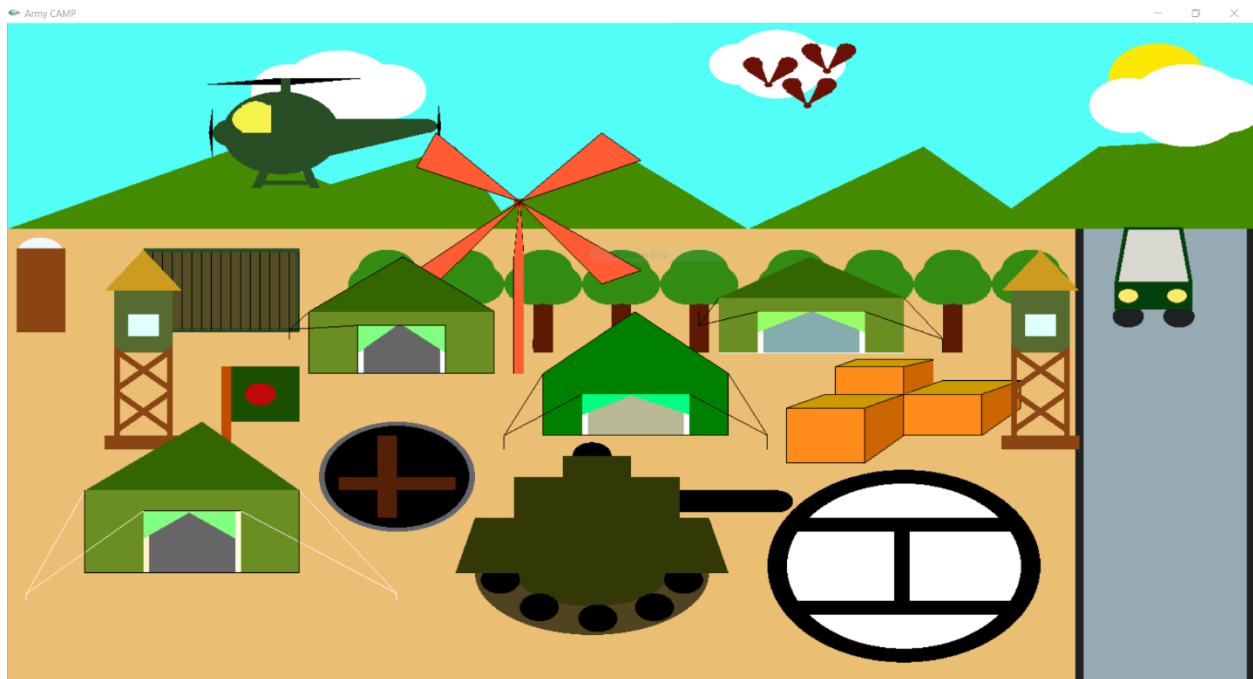
To execute this undertaking, we needed to investigate each little subtleties and gather them to expand proficiency of capacity utilization. For example, we can utilize the same arrangement of capacities for trees, vehicles, hills, tanks, cars, sun, moon etc can be drawn utilizing the same arrangement of capacities. Subsequent to gathering we concentrated some overabundance capacities which caused us to shape the undertaking. We likewise needed to clear our comprehension on graphical bends. Without graphical bends we were unable to draw haggles shapes. Additionally, we needed to become familiar with the scaling strategy for keeping up the goal of an overabundance screen. After that we picked the fitting capacity set that will draw the task things. Here are some methodologies that downscaled the implementation overview of some of the features:

1. Tent: for tent we have used a polygon and add a triangle above it. for polygon and triangle 4 and 3 points are needed. Then for the door again we have taken a polygon and the sides are divided into two more small size polygons recognized as door sides. After that , there are four more lines on both sides of the tent to tighten the tent . BY following this procedure 3 more tents have been added.
2. Box: for each box we have taken 1 polygon and 2 quads .Then add lines in each. This is how we have added 3 boxes.
3. HILL: for the hill we have taken 6 triangles. Each triangle consists of 3 corners. We have added one corner to the other.
4. windmill: for windmill we have taken one polygon which is the base of the windmill. then took a small sized triangle then added 4 more triangles as the fan of the windmill.

Significance:

- We used manual points throughout the whole project which helped us to understand the implementation to a project with the very basics.
- Have learnt about each function and how they work under circumstances.
- Have learnt how to beautify a view manually.
- Throughout the project, it is now easy to understand the basic operations of graphics. For example: translation, rotation, algorithms etc.
- After the completion of the project, a basic idea over designing is now understandable.

Project overview:





Conclusion:

The project has been completed through the very basic components and methods of OpenGL. The reason behind OpenGL is OpenGL is portable. It is a **cross**-platform API, i.e. it can be used on various platforms such as- Windows, Mac, and some handheld devices. It is more extensible, i.e. new **hardware** features are exposed quickly. It has a stable interface until some bigger changes are made.