

Turbine

1.0

Generated by Doxygen 1.10.0

1 File Index	1
1.1 File List	1
2 File Documentation	3
2.1 src/main.cpp File Reference	3
2.1.1 Detailed Description	4
2.1.2 Function Documentation	4
2.1.2.1 display()	4
2.1.2.2 drawFilledCircle()	4
2.1.2.3 drawTurbineStructure()	5
2.1.2.4 keyboard()	5
2.1.2.5 main()	5
2.1.2.6 readSpeedFromFile()	6
2.1.2.7 update()	6
Index	7

Chapter 1

File Index

1.1 File List

Here is a list of all documented files with brief descriptions:

src/ main.cpp	
Wind Turbine Simulation	3

Chapter 2

File Documentation

2.1 src/main.cpp File Reference

Wind Turbine Simulation.

```
#include <iostream>
#include <cmath>
#include <GL/glut.h>
#include <fstream>
#include <chrono>
#include <thread>
#include <string>
```

Functions

- `std::vector< float > readSpeedFromFile ()`
Read the Speed data from file.
- `void update (int value)`
Update function for blade rotation.
- `void drawFilledCircle (float cx, float cy, float r, int num_segments)`
Function to draw a filled circle.
- `void drawTurbineStructure ()`
Function to develop the structure of the turbine.
- `void display ()`
Display callback function.
- `void keyboard (unsigned char key, int x, int y)`
Keyboard callback function.
- `int main (int argc, char **argv)`
Main function.

Variables

- float **bladeRotationAngle** = 0.0f
Initial rotation angle.
- int **currentIndex** = 0
Index to get the speed from the file.
- std::vector< float > **SpeedData** = [readSpeedFromFile\(\)](#)
Array to store the speed data from the file.
- int **i** = 1
Index to denote Speed update.

2.1.1 Detailed Description

Wind Turbine Simulation.

Author

Sreenath Swaminathan

Date

03.02.2023

2.1.2 Function Documentation

2.1.2.1 display()

```
void display ( )
```

Display callback function.

This function is performs the rendering of the wind turbine. < Applying transformations and render the turbine structure

2.1.2.2 drawFilledCircle()

```
void drawFilledCircle (
    float cx,
    float cy,
    float r,
    int num_segments )
```

Function to draw a filled circle.

This function draws a filled circle using GL_TRIANGLE_FAN.

Parameters

<i>cx</i>	X-coordinate of the center
<i>cy</i>	Y-coordinate of the center
<i>r</i>	Radius of the circle
<i>num_segments</i>	Number of segments for circle approximation

2.1.2.3 drawTurbineStructure()

```
void drawTurbineStructure ( )
```

Function to develop the structure of the turbine.

This function creates the required structure to represent the WindTurbine This includes a Pole as its base A circular structure and blades rotating about the center of that circle. < Pole like structure

< Circle in the middle

< Rotating the object around the circle

< Blade 1 - Each blade is separated by almost 120 degrees

< Blade 2

< Blade 3

2.1.2.4 keyboard()

```
void keyboard (
    unsigned char key,
    int x,
    int y )
```

Keyboard callback function.

This function handles keyboard input.

Parameters

<i>key</i>	ASCII code of the pressed key
<i>x</i>	X-coordinate of the mouse when key was pressed
<i>y</i>	Y-coordinate of the mouse when key was pressed

< Handle keyboard input

2.1.2.5 main()

```
int main (
    int argc,
    char ** argv )
```

Main function.

Entry point of the program.

Parameters

<i>argc</i>	Number of command-line arguments
<i>argv</i>	Command-line argument values

Returns

int Exit status

Set up the viewport, projection, etc.

Register the display and reshape callback functions

Function to display the Wind turbine

Register the keyboard callback function

2.1.2.6 readSpeedFromFile()

```
std::vector< float > readSpeedFromFile ( )
```

Read the Speed data from file.

This function captures the speed data from 1 (Min) to 10 (Max) speed at which the turbine blades are to be rotated

Returns

std::vector<float> Array containing speed data

< Extract the speed value as a string

< Convert the speed value to a float

2.1.2.7 update()

```
void update (
    int value )
```

Update function for blade rotation.

This function updates the rotation angle of turbine blades based on the speed from the file. Speed is modified almost every 5 seconds.

Parameters

<i>value</i>	Timer value
--------------	-------------

< Updating the rotation speed

< Speed will be modified for every 5 secs in this case

< Schedule the next update

< Update function Progresses every millisecond

< Block that increments the speed after every 5 seconds

Index

- display
 - main.cpp, [4](#)
- drawFilledCircle
 - main.cpp, [4](#)
- drawTurbineStructure
 - main.cpp, [5](#)
- keyboard
 - main.cpp, [5](#)
- main
 - main.cpp, [5](#)
- main.cpp
 - display, [4](#)
 - drawFilledCircle, [4](#)
 - drawTurbineStructure, [5](#)
 - keyboard, [5](#)
 - main, [5](#)
 - readSpeedFromFile, [6](#)
 - update, [6](#)
- readSpeedFromFile
 - main.cpp, [6](#)
- src/main.cpp, [3](#)
- update
 - main.cpp, [6](#)