Experiment 1.1

Student Name: Ishant **UID: 22BCS15867**
Branch: CSE **Section:IOT-611**
Semester: 6th **DOP:8 JAN 2025**
Subject: Computer Graphics **Subject Code: 22CSH-352**

Aim: Demonstrate the use of graphics.h functions to draw basic shapes like lines, triangles, and circles.

Objective: To familiarize students with using the graphics.h library to create basic shapes like lines, triangles, and circles.

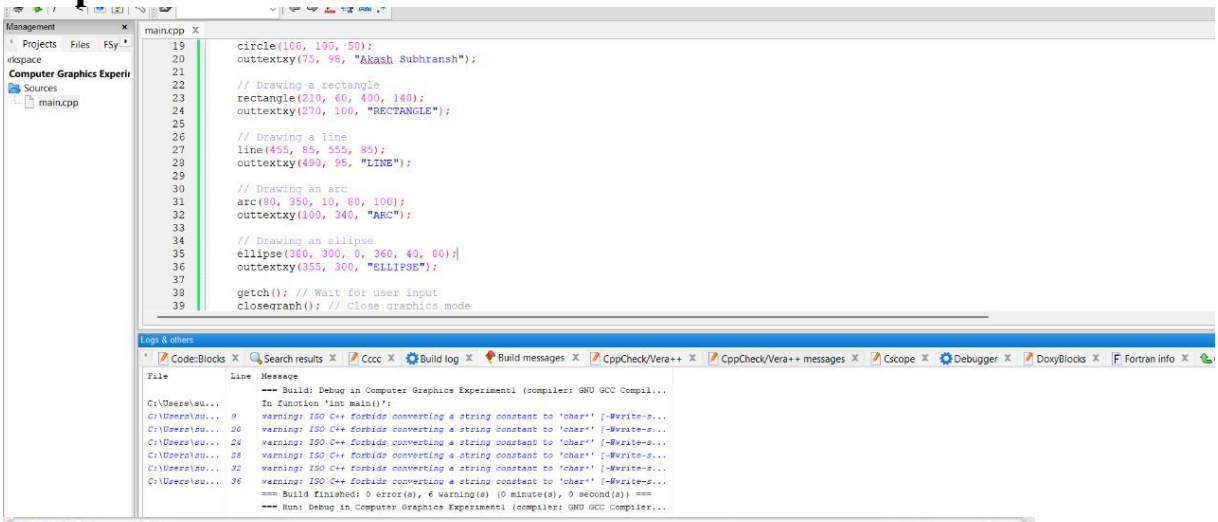
Algorithm:

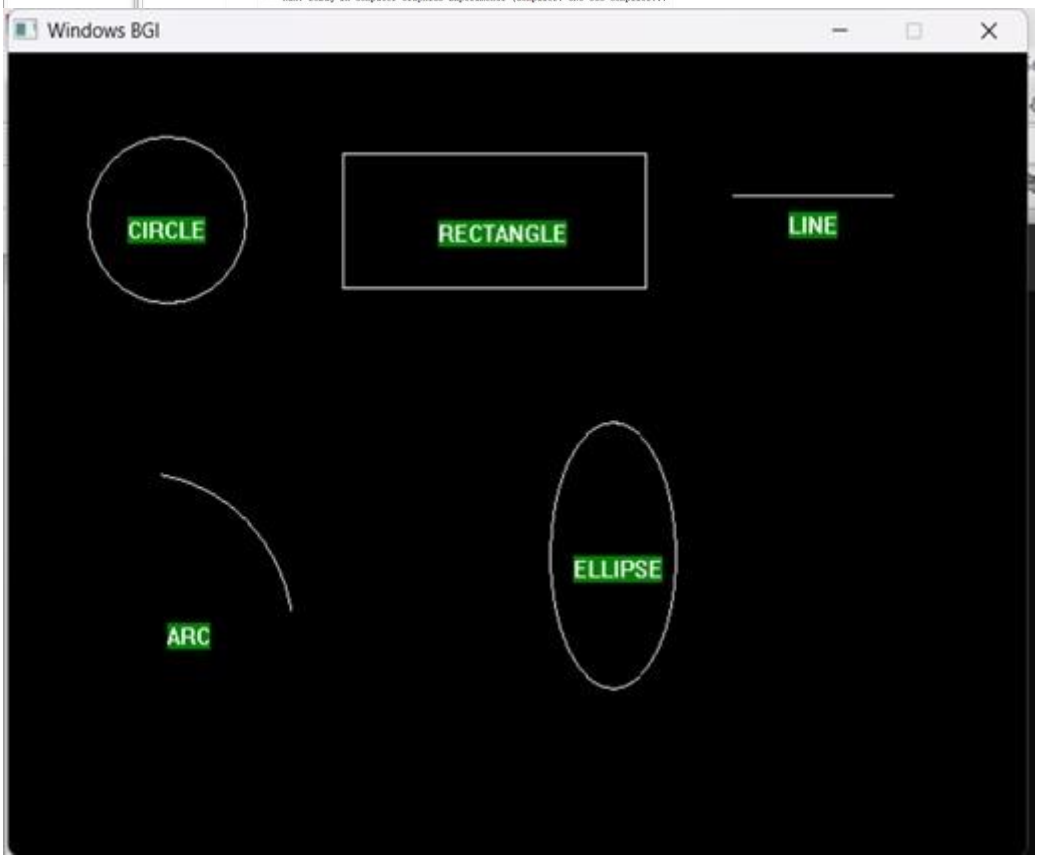
1. **Initialize Graphics Mode**:
   - Include the `graphics.h` header.
   - Use the `initgraph()` function to initialize the graphics mode.
2. **Draw a Line**:
   - Use the `line(x1, y1, x2, y2)` function to draw a straight line between two points `(x1, y1)` and `(x2, y2)`.
3. **Draw a Triangle**:
   - Use the `line()` function thrice to draw three sides of the triangle by connecting its vertices.
4. **Draw a Circle**:
   - Use the `circle(x, y, radius)` function to draw a circle with the center at `(x, y)` and a specified radius.
5. **Display Graphics**:
   - Use the `getch()` function to wait for a key press to keep the graphics window open.
6. **Close Graphics Mode**:
   - Use the `closegraph()` function to close the graphics mode and clean up resources.

Code:

```
#include<iostream.h>
#include<conio.h>
#include<graphics.h>

void main()
{
   clrscr(); // Clear screen
   int gd = DETECT, gm; // Graphics driver and mode
   initgraph(&gd, &gm, "C:\\TURBOC3\\BGI"); // Initialize graphics mode

   setbkcolor(2); // Set background color

   // Drawing a circle
   circle(100, 100, 50);
   outtextxy(75, 98, "CIRCLE");

   // Drawing a rectangle
   rectangle(210, 60, 400, 140);

 outtextxy(270, 100, "RECTANGLE");


  // Drawing a line
   line(455, 85, 555, 85);
   outtextxy(490, 95, "LINE");

   // Drawing an arc
   arc(80, 350, 10, 80, 100);
   outtextxy(100, 340, "ARC");

   // Drawing an ellipse
   ellipse(380, 300, 0, 360, 40, 80);
   outtextxy(355, 300, "ELLIPSE");

   getch(); // Wait for user input
   closegraph(); // Close the graphics mode
}
```

Output:


Management
Projects
Files
Computer Graphics Experi
Sources
main.cpp
main.cpp
19 circle(100, 100, 50);
20 outtextxy(75, 98, "Akash Subhransh");
21
22 // Drawing a rectangle
23 rectangle(210, 60, 400, 140);
24 outtextxy(270, 100, "RECTANGLE");
25
26 // Drawing a line
27 line(455, 85, 555, 85);
28 outtextxy(490, 95, "LINE");
29
30 // Drawing an arc
31 arc(80, 350, 10, 80, 100);
32 outtextxy(100, 340, "ARC");
33
34 // Drawing an ellipse
35 ellipse(360, 300, 0, 360, 40, 80);
36 outtextxy(355, 300, "ELLIPSE");
37
38 getch(); // Wait for user input
39 closegraph(); // Close graphics mode
Logs & others
Code::Blocks
Search results
Cccc
Build log
Build messages
CppCheck/Vera++
CppCheck/Vera++ messages
Cscope
Debugger
DoxyBlocks
Fortran info
File
Line
Message
=== Build: Debug in Computer Graphics Experiment1 (compiler: GNU GCC Compil...
In function 'int main()':
warning: ISO C++ forbids converting a string constant to 'char*' [-Wwrite-s...
warning: ISO C++ forbids converting a string constant to 'char*' [-Wwrite-s...
warning: ISO C++ forbids converting a string constant to 'char*' [-Wwrite-s...
warning: ISO C++ forbids converting a string constant to 'char*' [-Wwrite-s...
warning: ISO C++ forbids converting a string constant to 'char*' [-Wwrite-s...
warning: ISO C++ forbids converting a string constant to 'char*' [-Wwrite-s...
=== Build finished: 0 error(s), 6 warning(s) (0 minute(s), 0 second(s)) ===
=== Run: Debug in Computer Graphics Experiment1 (compiler: GNU GCC Compiler...


Windows BGI
CIRCLE
RECTANGLE
LINE
ELLIPSE
ARC

Learning Outcomes:

1. Utilize the graphics.h library in C++ to implement fundamental graphical elements.
2. Effectively initialize and terminate graphics mode in programs.
3. Demonstrate the creation of basic shapes, including lines, circles, rectangles, arcs, and ellipses.
4. Develop skills to position and display text relative to shapes using outtextxy.
5. Establish a foundation in computer graphics to support learning advanced topics.