**Lab Report. 06**

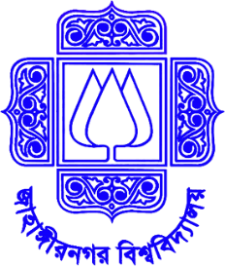
**Title: Lab Report**

*Course title: Computer Graphics Lab*

*Course code: CSE-304*

*3rd Year 1st Semester 2022*

**Date of Submission**: 13/08/2023



###### **Submitted to-**

###### Dr. Mohammad Shorif Uddin

###### Professor

*Department of Computer Science and Engineering*

*Jahangirnagar University*

*Savar, Dhaka-1342*

*And*

*Dr. Morium Akter*

###### Associate Professor

*Department of Computer Science and Engineering*

*Jahangirnagar University*

*Savar, Dhaka-1342*

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl** | Class Roll | Exam Roll | Name |
| 01 | 369 |  | Yumna Tasneem |

Liang Barsky Algorithm:

Source code:

|  |  |
| --- | --- |
| #include <iostream>  #include <graphics.h>  using namespace std;  void drawColoredLine(int x1, int y1, int x2, int y2, int color)  {  setcolor(color);  line(x1, y1, x2, y2);  setcolor(WHITE);  }  void liangBarsky(int x1, int y1, int x2, int y2, int xmin, int ymin, int xmax, int ymax)  {  int p[4], q[4];  int dx = x2 - x1, dy = y2 - y1;  p[0] = -dx;  q[0] = x1 - xmin;  p[1] = dx;  q[1] = xmax - x1;  p[2] = -dy;  q[2] = y1 - ymin;  p[3] = dy;  q[3] = ymax - y1;  float u1 = 0, u2 = 1;  for (int i = 0; i < 4; i++)  {  if (p[i] == 0)  {  if (q[i] < 0)  {  return; // Line lies completely outside the window  }  }  else  {  float t = (float)q[i] / p[i]; | if (p[i] < 0)  {  u1 = max(u1, t);  }  else  {  u2 = min(u2, t);  }  }  }  if (u1 > u2)  {  return; // Line lies completely outside the window  }  int clippedX1 = x1 + u1 \* dx;  int clippedY1 = y1 + u1 \* dy;  int clippedX2 = x1 + u2 \* dx;  int clippedY2 = y1 + u2 \* dy;  drawColoredLine(x1, y1, clippedX1, clippedY1, RED);  drawColoredLine(clippedX1, clippedY1, clippedX2, clippedY2, GREEN);  drawColoredLine(clippedX2, clippedY2, x2, y2, RED);  }  int main()  {  int gd = DETECT, gm;  initgraph(&gd, &gm, "");  int x1 = 50, y1 = 50, x2 = 300, y2 = 250;  int xmin = 100, ymin = 100, xmax = 400, ymax = 300;  rectangle(xmin, ymin, xmax, ymax);  liangBarsky(x1, y1, x2, y2, xmin, ymin, xmax, ymax);  getch();  closegraph();  return 0;  } |

Output:

