Indian Institute of Technology, Kharagpur

CS19001 Programming and Data Structures Laboratory, Section 10

Spring 2020

# Lab Test 2 (May 5, 2020)

**Total Marks:** 20                                                              **Duration:** 2 hrs

**INSTRUCTIONS** : Submit a single C file named **[rollno]-lseg3.c** where '[rollno]' is your roll number.

1.  (a) Define the data type `Pointf` for a point (Point) in 2-d plane. The co-ordinates are floating point numbers. (3)

    (b) Define a data type `Lineseg` for a line segment. A line segment may be represented by its two end points.

    (c) Define a structure `Pointlist` that has two fields: a pointer to type Pointf and the number of points.

2.  Implement the following functions:

    (a) `int intersects (Lineseg l1, Lineseg l2)` : returns 1 if l1 and l2 intersect, 0 otherwise (2)

    (b) `Pointf intersectpoint (Lineseg l1, Lineseg l2)` : returns the point of intersection of l1 and l2 (2)

    (c) `Lineseg genls ( )` : Generate a random line segment in the plane with end points between $(0, 0)$ and $(1, 1)$. Returns the structure containing the line segment. (2)

    (d) `void printls ( Lineseg lserg)` : Print the line segment lseg. (1)

    (e) `void printall (Pointlist plist )` : Print all points in the list (1)

    (f) `int countintpt (Lineseg llist [])` : Returns the number of intersection points of the line segments. You may find the number of pairs of line segments that intersect. You need not check whether three or more line segments intersect at the same point. (2)

    (g) `void createplist ( Lineseg llist [] , Pointlist * pplist, int num)` : Allocate memory to store num points in Plist via its pointer pplist. Populate the structure Plist through its address pplist by finding the points of intersection of line segments in llist and including them within Plist. (4)

3. Write a main ( )                                                                (3)

    (a) Declare a variable Plist of type Pointlist

    (b) Declare LSarr as an array of 10 Lineseg.

    (c) Generate 10 random line segments and store them in the above array.

    (d) Find the number of points of intersection where two of the line segments intersect. Let it be num.

    (e) Allocate memory to store num points in Plist.

    (f) Find the points of intersection again and store them in Plist.

    (g) Print all the point in Plist

You must call the above functions to do this.

**Note:**

To generate random numbers use: **rand()**. This returns a random number (integer) between 0 to RAND_MAX (which is at least 32767) The constant RAND_MAX is defined in standard library (stdlib.h). You have to convert the integer random number returned by rand () to a floating point number between 0 and 1.