

## ESC101 : Fundamental of Computing

Lab 7 for 23rd September 2008

**Hint :** The classes for **Point**, **Triangle**, **Circle** are available along with the sample programs on the course website. In this lab exercise, you might like to go through these classes and sample programs.

An axis parallel rectangle is a rectangle whose sides are parallel to x-axis and y-axis. It is easy to observe that an axis parallel rectangle is uniquely specified by coordinates of its lower left corner and upper right corner.

1. (Marks = 5)

Design and implement a class **Axis\_parallel\_rect** (using the class **Point**) with the following attributes and methods.

**Attributes :**

`Point lower_left_corner;`

`Point upper_left_corner;`

**Constructors :**

- **Axis\_parallel\_rect(Point L, Point U)**

It should build an axis parallel rectangle with lower left corner same as that of the Point referenced by L and upper right corner as the Point referenced by U.

- **Axis\_parallel\_rect(Point L, double x\_length, y\_length)**

It should build an axis parallel rectangle with lower left corner same as that of Point referenced by L, length along x-axis equal to as **x\_length** and length along y-axis equal to **y\_length**.

- **Axis\_parallel\_rect()**

It should build the axis parallel rectangle with lower left corner as well as upper right corner at the origin (0,0).

**Methods :**

- (a) `public double Area()`

To determine area of the axis parallel rectangle.

- (b) `public double Perimeter()`

To determine perimeter of the axis parallel rectangle.

- (c) `public Point Center()`

To return the center of the axis parallel rectangle.

- (d) `public boolean is_inside(Point P)`

to determine if a point P is located inside the axis parallel rectangle.

2. (marks = 5)

(**Note :** You have to use the class **Axis\_parallel\_rect** described above in this part of the exercise.)

First develop two methods.

- `public boolean Intersect(Axis_parallel_rect R1, Axis_parallel_rect R2)`  
to determine if axis parallel rectangles R1 and R2 intersect.
- `public boolean Intersect(Axis_parallel_rect R, Triangle T)`  
to determine if axis parallel rectangle R intersects triangle T.

You have to develop the above methods using the methods available from class `Axis_parallel_rect` or `Point`. Use these two methods to develop a program which will take as input the coordinates of lower left corner and upper right corner of an axis parallel rectangle from command-line. Let us denote this triangle with name *A*. The program should prints the the following output with suitable messages.

- Area of Rectangle A :
- Perimeter of Rectangle A :
- Length and width of the rectangle A :
- Center of rectangle A :
- Whether origin lies inside A ?
- Whether the triangle with endpoints :  $\{(1,0)(-1,0)(0,1)\}$  intersects the rectangle A ?
- Whether the axis parallel rectangle  $\{(0,0), (5,5)\}$  intersects the rectangle A ?