ESC101 : Fundamental of Computing
Lab 7 for 29th September 2008

1. (Marks = 5)
A one dimensional open interval (ℓ, r) for real numbers ℓ, r with $\ell \leq r$ is the set of all those points p on the real line such that $\ell < p < r$. Here we call ℓ as the *left* endpoint and r as the *right* endpoint of the interval (ℓ, r). An interval (ℓ, r) is called empty if $\ell = r$. It can be seen that an empty interval does not contain any point of the real line.

Design and implement a class Interval with the following attributes and methods.
Attributes :
```
double left;
double right;
```
Constructors :

- `Interval(double ℓ, double r)`
  It should build an interval with left endpoint = $\min(\ell, r)$ and right endpoint = $\max(\ell, r)$.
- `Interval(double ℓ)`
  It should build an interval with left endpoint ℓ and right endpoint $\ell + 1$.
- `Interval( Interval I)`
  It should build a copy of the interval I, that is, left endpoint of the new interval should be the same as the left endpoint of I, and the right endpoint of the new interval should be the same as the right endpoint of I.

Methods :

(a) `public double Length()`
To determine the length of the current interval. The length of an interval is defined in the usual manner. For example, length of the interval $[2.5, 3.8]$ is 1.3 and length of the interval $[-4, 2]$ is 6.

(b) `public boolean Isempty()`
Returns true if the current interval is empty.

(c) `public boolean Contains( double r)`
Returns *true* if the current interval contains the point r of real line, and false otherwise. For example, $[2, 5.4]$ contains the point 3.1 but does not contain the point 6.

(d) `public Interval intersection(Interval I)`
To return the reference to an interval which is the intersection of the current interval and the interval I. The intersection is defined in the usual manner. For example, intersection of interval $(1, 4.3)$ with the interval $(2, 6)$ is equal to the interval $(2, 4.3)$, whereas Intersetion of interval $(1, 4.3)$ with interval $(-4, 0)$ is an *empty* interval. We shall follow the convention that if (ℓ_1, r_1) does not intersects (ℓ_2, r_2), and $r_1 \leq \ell_2$, then the intersection interval is (r_1, r_1).

(e) `public double MidPoint()`
Returns the mid point of the current interval. For example, mid point of the interval $(1, 4.4)$ is 2.7.

(f) `public double Distance(Interval I)`
Returns the distance between the current interval and the interval I. The distance between two intervals is equal to the absolute difference between their mid-points. This method **must** invoke the MidPoint method desribed above. Note that the distance is always non-negative.

2. (marks = 5)

Use Interval class to develop a program which will take as input the endpoints of two intervals I_1 and I_2 from command line. (Total four command line arguments).

For examples, if the arguments are 3 1 2 4, then $I_1 = (1, 3)$ and $I_2 = (2, 4)$. The program should compute the following information about these intervals and print suitable messages.

- The length of Intervals I_1 and I_2 respectively.
- The interval which is bigger, that is, the interval whose length is greater than the other.
- Whether any of the intervals I_1 and I_2 is empty.
- Whether any of the intervals contains the origin.
- Whether the intervals I_1 and I_2 intersect, if yes, print the the interval which is intersection of these two intervals.
- Print the distance between the intervals I_1 and I_2.