Create a method getDistance(a, b) that calculates the distance between a and b.

Input

```
import java.util.HashMap;

import java.util.Map;

import java.util.PriorityQueue;

public class PrioRITYQueue Example {

public static void main(String[] args){

PriorityQueue<Double> pq = new PriorityQueue<Double>((x,y)-> {Double z =
y-x;return z.intValue(); });

PrioRQueueExample pqe = new PrioRITY Queue Example();

//Number of ATMs to return i.e. K

int num_ATMs = 3;

double curr_loc = 0.00;

Map<String,Double> nallAT Locs = new HashMap<String,Double>();

//Map of ATM names and their distance coordinates

nallATMLocs.put("atm1",45.0);

nallATMLocs.put("atm2",78.0);

nallATMLocs.put("atm3",54.0);

nallATMLocs.put("atm4",64.0);
nallATMLocs.put("atm5",35.0);

nallATMLocs.put("atm6",42.0);

nallATMLocs.put("atm7",57.0);

nallATMLocs.put("atm7",1.00);

nallATMLocs.forEach((atm,dist) ->{if(pq.size() < num_ATMs){

pq.add(pqe.getLocation(curr_loc,dist));}

else{

if(        pq.peek() > pqe.getLocation(curr_loc,dist)){

pq.poll();

pq.add(pqe.getLocation(curr_loc,dist));

}

}

});

pq.forEach(atm Lock -> System.out.println(atmLoc));

}

private double getLocation(double curr,double atm){

return atm - curr;

}
```

}

OutPut

```
 $javac PrioRQueueExample.java
 $java -Xmx128M -Xms16M PrioRQueueExample
42.0
1.0
35.0
```