Baris Ayyildiz 1901042252 CSE 222 Final Examination B.Ayyildiz

I hereby pledge on my honor that I will strictly adhere to academic integrity codes and the work done on this examination is solely my own and I will not receive/give any help from/to anybody or source during this examination.

4 min

```
public class GraphHash implements Graph {
  private class Edge {
    private int source;
    private int dest;
    private double weight;
    public Edge(int s,int d){
      this.source = s;
      this.dest=d; this.weight=1.0;
    }
    public Edge(int s,int d,
                   double w){
      this.source=s;
      this.dest=d;
      this.weight=w;
    }
  }
  private HashSet<Edge> hSet;
  public GraphSet(int numV);
    this.hSet= new HashSet<Edge>();
    this.numV =numV;
  }
  public void add(int source,int dest){
    Edge temp= new Edge(source,dest);
    if(!hSet.contains(temp)) hSet.add(temp));
  }
  public void remove(int source,int dest){
    Edge temp= new Edge(source,dest);
    if(hSet.contains(temp)) hSet.remove(temp);
  }
  public boolean hasEdge(int source,int dest){
    return hSet.contains(new Edge(source,dest));
  }
```

Barıs Ayyıldız 1901042252 CSE 222 Final Examination B.Ayyıldız

```
public Edge edge(int source, int dest){
  Edge temp = new Edge(source,dest);
  if(hSet.contains(temp)) return temp;
  return null;
}
public int numV(){return numV;}

private class HashGraphIterator<Edge> implements Iterator<Edge>{
  private int source;
  private int cursor;
  public HashGraphIterator(int source){
     this.source=source;
     this.cursor=0;
     adjustCursor();
  }
  private void adjustCursor(){
    while(cursor< numV()&& !hSet.contains(new Edge(source,
                                                cursor))){
      cursor++;
    }
  }
  public boolean hasNext(){return cursor != numV();}
  public Edge next(){
     Edge temp= new Edge(source, cursor);
     adjustCursor();
     return temp;

  }

public Iterator<Edge> edgeIterator(int source){
   return new HashGraphIterator(source);
}
```