**Resolução do Exame Recurso de ESINF (09 de Fevereiro 2017)**

Nelson Tavares Rafael (1101369)

1.

Com aritmética de índices resolve-se facilmente.

posant<-3

numaleatorio<-6

posicao<-3+6=9-8=1

Resolução Exemplo: (Bruno Fonseca)

Map<Integer, LinkedList<Integer>> roletaRussa(LinkedList<Integer> list){

Map<Integer,LinkedList<Integer>> mapLists = new LinkedHashMap<>();

LinkedList<Integer> cList = list.clone();

Random generator = new Random();

Integer p = 0, rand = 0;

While(!cList.isEmpty()){

rand = generator.nextInt(cList.size());

p = p + rand;

If ( p > cList.size())

p = p - cList.size();

CList.remove(p);

mapLists.put(p,cList.clone);

}

return mapLists;

}

2.

Remove o mínimo da heap.

3.

Adaptar o find.

Para obter a profundidade do caminho:

level <-level+1

depth<-depth+1

Resolução Exemplo: (Bruno Fonseca)

public Integer rootToNode(E e, LinkedList<E> list){

int p = 0;

return rootToNode(this.root, e, list, p);

}

private Integer rootToNode(Node<E> node, E e, LinkedList<E> list, p){

if(node == null){

list = null;

Return -1;

}

E elemt =node.getElement();

list.addLast(elemt);

if(elemt.compareTo(e) == 0)

return p;

if(elemt.compareTo(e) > 0)

return p + rootToNode(node.getLeft(), e, list, p+1);

else

return p + rootToNode(node.getRight(), e, list, p+1);

}

4.

Usar o allPaths, menor caminho em termos de ligações e usar o getEdge.

5.

Tem o mesmo número de elementos (primeira condição)

Faz-se removeMin num ciclo.