|  |
| --- |
| TAD <Max priority queue > |
| Max priority queue={size,comparator} |
| Inv: {comparator(a,b)= True} |
| Primitive Operations:  Createpriorityqueue(size): ->priorityqueue  enqueue(dato) : priorityqueuexnodo -> void  Dequeue(): priorityqueue->Node  peek (): -> node  size(): -> data  Clear (): -> void |

|  |
| --- |
| TAD <Heap> |
| Heap={parent,left,right,size} |
| Inv: {parent>left ^ parent>right } |
| Primitive Operations:  CreateHeap(size): ->Heap  InsertHeap(dato) : Heapxnode -> void  DeleteHeap (): Heap->Node  Find (): -> node  sizeHeap(): -> data  isEmpty(): -> boolean |

|  |
| --- |
| TAD <Hash table > |
| Hash table={size,hashfunction, keyequalityfunction,table} |
| Inv: { key ≠ table(keys)} |
| Primitive Operations:  createHashTable(size, hashFunction, keyEqualityFunction): ->Hashtable  put( key, value): Hashtablexkey^value -> void  get(key): Hashtable->value  remove(key): Hashtablexkey->boolean  containsKey(key) :Hashtablexkey->boolean  sizeHeap(): -> data  isEmpty(): -> boolean  Clear (): -> void |