| **TAD Colas de prioridad** |
| --- |
|  |
| {inv: (elements={e1, e2,..en}∧ priorities={p1,p2,...pn} ⩝ pi ⩾pj ⟷ ei ﹥prioridad que ej →1≤ i ∧ j ≤ n) }  {inv: (elements:{A,B ,C, D} ∧ priorities={3, 6, ,5, 1}= 3≤5≤,6≤1)} |
| * PQueue() →PQueue() * insert() e→PQueue() →PQueue() * extract() PQueue()→e →e * front() PQueue()→e →e * isEmpty() PQueue()→e →empty or not PQueue() |
| PQueue()(Constructor)  “Crea una cola de prioridad vacía”  {pre: PQueue !=null}  {post PQueue pq=0:} |
| insert()(Modificador)  “Inserta un nuevo elemento <e> al final de la cola”  {pre:PQueue pq= (e1,e2,...en) ∧ element e ∧ q=0 }  {post:PQueue pq=(e1,e2,...en)} |
| extract()(modificador)  “Extrae un elemento de la cola”  {pre:PQueue pq ≠0 ∨ pq=(e1,e2,...en)}  {post:PQueue pq= (e1,e2,...en-1)} |
| front()(analizador)  “Da el valor que está enfrente de la cola”  {pre: PQueue pq=(e1,e2,...en)}  {post: e1} |
| isEmpty()(analizador)  “Define si la cola está vacía”  {pre: PQueue pq}  {post:If pq ≠ 0→ FALSE; else (es decir, pq=0)→ TRUE } |

| **TAD Colas** |
| --- |
| Queue=[(e1,e2,...en), front, back] |
| {inv: 0≤n ∧ size(Queue)=n ∧ front= e1 ∧ back=en} |
| * Queue() → Queue() * enqueue() Queue()→e → Queue() * dequeue() Queue()→e → e * front() Queue()→e → e1 * isEmpty() Queue() → TRUE ∨ FALSE |
| Queue() (Constructor)  “Crea una cola vacía”  {pre: Queue !=null}  {post: Queue=0} |
| enqueue() (Modificador)  “Inserta un elemento <e> al final de la cola <q>”  {pre: Queue q= (e1,e2,...en) ∧ e ∨ q=0 ∧ e}  {post: Queue q= (e1,e2,...en+1)} |
| dequeue()(Modificador)  “Extrae un elemento <e> del frente de la cola <q>”  {pre: Queue q=0 ∨ q=(e1,e2,...en)}  {post:Queue q=(e2,e3...en-1)∧ e1 } |
| front()(Analizador)  “Muestra el valor de <e> que esté al frente de la cola <q>”  {pre: Queue q=0 ∨ q=(e1, e2,...en)}  {post: e1} |
| isEmpty()(Analizador)  “Determina si la cola <q> está vacía o no”  {pre: Queue q}  {post: TRUE if q=0, FALSE if q≠0} |