

## Prioritásos sor típus

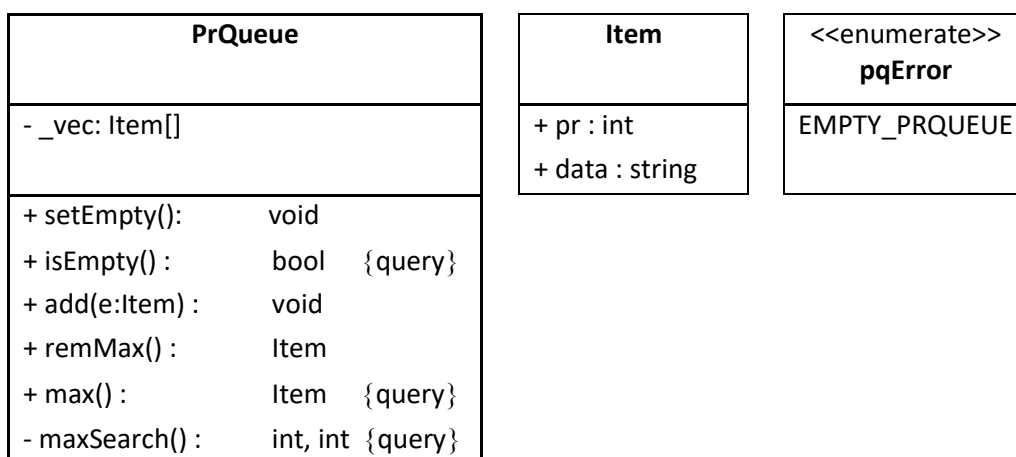
### Prioritásos sor

|  |   |                  |  |      |  |                            |                                     |               |                        |                      |                          |  |                  |  |      |  |                            |                                     |               |
|--|---|------------------|--|------|--|----------------------------|-------------------------------------|---------------|------------------------|----------------------|--------------------------|--|------------------|--|------|--|----------------------------|-------------------------------------|---------------|
| <p><b>Típus értékek:</b></p> <p>PrQueue a maximum prioritásos sorok halmaza, amely soroknak az elemei <math>\mathbb{Z} \times \mathbb{S}</math> típusú párok.</p>                      | <p><b>Típus műveletek:</b></p> <p><b>pq.setEmpty()</b> <span style="float: right;">pq : PrQueue</span><br/> <i>// Kiüríti a pr sort</i></p> <p><b>l := pq.isEmpty()</b> <span style="float: right;">pq : PrQueue, l : <math>\mathbb{L}</math></span><br/> <i>//Igazat ad, ha üres a pr sor, hamisat ha nem.</i></p> <p><b>pq.add(e)</b> <span style="float: right;">pq : PrQueue, e : <math>\mathbb{Z} \times \mathbb{S}</math></span><br/> <i>//Beteszi egy új elemet a pr sorba.</i></p> <p><b>e := pq.remMax()</b> <span style="float: right;">pq : PrQueue, e : <math>\mathbb{Z} \times \mathbb{S}</math></span><br/> <i>//Kiveszi az egyik legnagyobb prioritású elemet. Fontos, hogy a sor nem lehet üres.</i></p> <p><b>e := pq.max()</b> <span style="float: right;">pq : PrQueue, e : <math>\mathbb{Z} \times \mathbb{S}</math></span><br/> <i>//Visszadja az egyik legnagyobb prioritású elemet, nem veszi ki. Fontos, hogy a sor nem lehet üres.</i></p>   |                  |  |      |  |                            |                                     |               |                        |                      |                          |  |                  |  |      |  |                            |                                     |               |
| <p><b>Típus reprezentáció:</b></p> <p>vec: Item[] – prioritásos sor elemeit rendezetlen tároló tömb, ahol Item = rec(pr : <math>\mathbb{Z}</math>, data : <math>\mathbb{S}</math>)</p> | <p><b>Típusműveletek implementációja:</b></p> <p><b>pq.setEmpty()</b><br/> <span style="border: 1px solid black; padding: 2px;">vec := &lt;&gt;</span> (C++ vector: clear())</p> <p><b>l := pq.isEmpty()</b><br/> <span style="border: 1px solid black; padding: 2px;">l :=  vec  = 0</span> (C++ vector: size())</p> <p><b>pq.add(e)</b><br/> <span style="border: 1px solid black; padding: 2px;">vec := vec <math>\oplus</math> &lt;e&gt;</span> (C++ vector: push_back())</p> <p><b>e, err := pq.remMax()</b></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2" style="text-align: center;">err := ( vec =0)</td> </tr> <tr> <td colspan="2" style="text-align: center;">-err</td> </tr> <tr> <td style="width: 70%;">max, ind := maxSearch(vec)</td> <td rowspan="4" style="text-align: center; vertical-align: middle;"><b>Hiba:</b> üres a prioritásos sor</td> </tr> <tr> <td>e := vec[ind]</td> </tr> <tr> <td>vec[ind] := vec[ vec ]</td> </tr> <tr> <td>vec:=vec[1.. vec -1]</td> </tr> <tr> <td colspan="2" style="text-align: center;">(C++ vector: pop_back())</td> </tr> </table> <p><b>e, err := pq.max()</b></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td colspan="2" style="text-align: center;">err := ( vec =0)</td> </tr> <tr> <td colspan="2" style="text-align: center;">-err</td> </tr> <tr> <td style="width: 70%;">max, ind := maxSearch(vec)</td> <td rowspan="2" style="text-align: center; vertical-align: middle;"><b>Hiba:</b> üres a prioritásos sor</td> </tr> <tr> <td>e := vec[ind]</td> </tr> </table> | err := ( vec =0) |  | -err |  | max, ind := maxSearch(vec) | <b>Hiba:</b> üres a prioritásos sor | e := vec[ind] | vec[ind] := vec[ vec ] | vec:=vec[1.. vec -1] | (C++ vector: pop_back()) |  | err := ( vec =0) |  | -err |  | max, ind := maxSearch(vec) | <b>Hiba:</b> üres a prioritásos sor | e := vec[ind] |
| err := ( vec =0)   |   |                  |  |      |  |                            |                                     |               |                        |                      |                          |  |                  |  |      |  |                            |                                     |               |
| -err   |   |                  |  |      |  |                            |                                     |               |                        |                      |                          |  |                  |  |      |  |                            |                                     |               |
| max, ind := maxSearch(vec)   | <b>Hiba:</b> üres a prioritásos sor   |                  |  |      |  |                            |                                     |               |                        |                      |                          |  |                  |  |      |  |                            |                                     |               |
| e := vec[ind]  |   |                  |  |      |  |                            |                                     |               |                        |                      |                          |  |                  |  |      |  |                            |                                     |               |
| vec[ind] := vec[ vec ]   |   |                  |  |      |  |                            |                                     |               |                        |                      |                          |  |                  |  |      |  |                            |                                     |               |
| vec:=vec[1.. vec -1]   |   |                  |  |      |  |                            |                                     |               |                        |                      |                          |  |                  |  |      |  |                            |                                     |               |
| (C++ vector: pop_back())   |   |                  |  |      |  |                            |                                     |               |                        |                      |                          |  |                  |  |      |  |                            |                                     |               |
| err := ( vec =0)   |   |                  |  |      |  |                            |                                     |               |                        |                      |                          |  |                  |  |      |  |                            |                                     |               |
| -err   |   |                  |  |      |  |                            |                                     |               |                        |                      |                          |  |                  |  |      |  |                            |                                     |               |
| max, ind := maxSearch(vec)   | <b>Hiba:</b> üres a prioritásos sor   |                  |  |      |  |                            |                                     |               |                        |                      |                          |  |                  |  |      |  |                            |                                     |               |
| e := vec[ind]  |   |                  |  |      |  |                            |                                     |               |                        |                      |                          |  |                  |  |      |  |                            |                                     |               |

A több művelet is használja a  $(\max, \text{ind}) = \text{MAX}_{i=1..|\text{vec}|} (\text{vec}[i].\text{pr})$  maximum kiválasztást.

## Prioritásos sor típus

### UML diagrammok:



### Egység teszt példa:

| remMax() tesztelése                          |            |                            |  |
|--|------------|----------------------------|--|
| teszteset                                    | teszt tömb | eredmény                   | tömb új tartalma   |
| üres intervallum                             | <>         | hiba<br>(kivétel dobás)    | <>   |
| egy elemű                                    | <3>        | 3                          | <>   |
| több elemű esetek:                           |            |                            |  |
| első a legnagyobb                            | <5,2,3>    | 5                          | <3,2>  |
| utolsó a legnagyobb                          | <1,2,3>    | 3                          | <1,2>  |
| belső a legnagyobb                           | <1,3,2>    | 3                          | <1,2>  |
| nem egyértelmű, első és utolsó a legnagyobb  | <5,2,5'>   | 5                          | <5',2> (az adat rész segítségével ellenőrizhető, hogy az elsőt vettük ki)                                  |
| nem egyértelmű, belső és utolsó a legnagyobb | <1,3,3'>   | 3                          | <1,3'>   |
| mind egyforma                                | <3,3',3''> | 3                          | <3'',3'>   |
| több egymás utáni remMax(), majd add hatása  | <2,3,1>    | 3<br>2<br>1<br>3<br>2<br>1 | <2,1> remMax()<br><1> remMax()<br><> add(3) add(2) add(1) remMax()<br><1,2> remMax()<br><1> remMax()<br><> |