Heap operations in Maple

```
> S := [[x,5], [x,6], [y,2], [x,1], [x,3], [z,7], [x,2], [y,5]];
                S := [[x, 5], [x, 6], [y, 2], [x, 1], [x, 3], [z, 7], [x, 2], [y, 5]]
```

Sort S on the second component

```
> gtr := proc(a,b) evalb( a[2] < b[2] ) end;
                    gtr := proc(a, b)  evalb(a[2] < b[2])  end proc
```

```
> H := heap[new](gtr);
                           H := table([0 = 0, `<` = gtr])
```

```
> for a in S do heap[insert](a,H) od;
                                      [x, 5]
                                      [x, 6]
                                      [y, 2]
                                      [x, 1]
                                      [x, 3]
                                      [z, 7]
                                      [x, 2]
                                      [y, 5]
```

```
> heap[max](H);
                                      [z, 7]
```

```
> while not heap[empty](H) do heap[extract](H) od;
                                      [z, 7]
                                      [x, 6]
                                      [y, 5]
                                      [x, 5]
                                      [x, 3]
                                      [x, 2]
                                      [y, 2]
                                      [x, 1]
```