

Selection Sort Algorithm

This is to help you with programming exercise 3, on page 37: the pseudocode of Selection Sort

```
procedure SelectionSort(items):
    Sorted = 0 # index of sorted elements
    n = number of elements in the list
    while sorted < n-2: # we need to sort the first n-1 elements only
        m = FindSmallest(items[sorted:]) # find next smallest
        # swapping the next min value with the element at current position
        tmp = items[sorted]
        items[sorted]= items[m+sorted]
        items[m+sorted]=tmp

        sorted += 1 # incrementing current position index, the number of
sorted elements, in the list.
```

We need to define the procedure of finding the smallest element in the list. It can be done by a built-in function call, or by defining a new function/procedure/method.

```
procedure FindSmallest(items):

    s = 0 # index of smallest element
    i = 0 # index/counter
    while i < len(items): # check all the elements in the list
        if items[s] > items[i]:
            s = i
        i +=1
    return s
```

Another option is to use the built-in **min** method. This will yield a better running time!