University College Cork
Computer Science
Department

Programming Training

Main Points:

- More Fundamental Algorithms on Arrays.

The counting problem

The counting problem:

If a=[a[i], i=0,1,…,n-1] is an array / list with n elements find the number of elements that satisfy a condition ***C***.

Inputs: a=(a[i], i=0,1,…,n-1) – list with n elements

Output: nr – int

How to do it:

Step 1. (initialization) nr=0;

Step 2. (repetition) repeat for each index in range 0,1,2,…,n-1

test if a[i] satisfies *C* and then increase nr;

COUNT HOW NAMY NUMBERS ARE POSITIVE

```
def arrayCount(a):
    # initialise nr
    nr = 0

    # traverse the list
    for i in range(len(a)):
        # test condition for a[i]
        if a[i]> 0 :
            nr = nr +1
        #endif
    #endfor

    return nr
#end def
```

The searching problem

The searching problem:

If a=[a[i], i=0,1,…,n-1] is an array / list with n elements find whether the element *X* is in the array and find its position.

Inputs: a=[a[i], i=0,1,…,n-1] – the list with n elements

Output: pos – int

How to do it:

Step 1. (initialization) pos=-1;

Step 2. (repetition) repeat for each index in range 0,1,2,…,n-1

test if a[i]==X then record the position pos=i;

```
def arraySearch(x, a):
    # initialise pos
    pos = -1

    # traverse the list
    for i in range(len(a)):
        # test if x is equal to  a[i]
        if a[i] == x :
            pos = i
            break
        #endif
    #endfor

    return nr
#end def
```

Note that pos == -1 when X does not belong to X.

Also note that breaking the repetition find the first occurrence

The Sorting problem

The Sorting problem:

If a=[a[i], i=0,1,…,n-1] is an array with n elements then reorganise the elements so that a[0] < a[1]<…<a[n-1].

Swap two variables a and b using a tmp:

tmp=a; a=b; b=tmp;

Compare and Exchange two elements a and b. Denote this operation as a↔b

```
if(a>b)
{
    tmp=a; a=b; b=tmp;
}
```

The Sorting problem

Inputs: a=[a[i], i=0,1,…,n-1] – the initial list

Output: a=[a[i], i=0,1,…,n-1] – the sorted list

How to do it:

repeat the following chains of compare and exchange:

a[0] ↔a[1] ↔a[2] ↔ … ↔a[n-2] ↔a[n-1]

a[0] ↔a[1] ↔a[2] ↔ … ↔a[n-3] ↔a[n-2]

….

a[0] ↔a[1] ↔a[2]

a[0] ↔a[1]

The Sorting problem

How to do it:

repeat for i=n-1, n-2, …, 1

// a[0] ↔a[1] ↔a[2] ↔ … ↔a[i]

repeat for j=0, 1, …,i-1

a[j] ↔a[j+1]

```
def bubbleSort(a):

    n = len(a)

    for i in range(n-1, 0, -1):
        for j in range(i-1):
            if a[j]>a[j+1] :
                a[j], a[j+1] = a[j+1], a[j]
            #endif
        # endfor
    #endfor

    return

#end def
```

To do List

1. Solve the HW problems.