CMPS 10 1-22-08

INPUT: $n \geq 1$ (# of #s in list)
$a_1, \ldots, a_n$ (the list items)

OUTPUT: the largest value in the list, and the position where it is located.

ASSUME: the #s in the list are distinct.

FIND LARGEST

```
1.) max ← a_1
2.) j ← 1
3.) i ← 2
4.) while i ≤ n
5.)     if a_i > max
6.)         max ← a_i
7.)         j ← i
8.)     i ← i + 1
9.) Print max
10.) Print j
11.) stop
```

```
1.) max ← a_1
2.) j ← 1
3.) for i ← 2 to n
4.)     if a_i > max
5.)         max ← a_i
6.)         j ← i
7.) Print max, j
8.) stop
```

EXERCISE: TRACE THIS ALGORITHM ON:

2, 1, -7, 5, 3, 11, 0

EXERCISE: WHAT HAPPENS IF WE REPLACE

> BY ≥ IN LINE 5. (ASSUME NOW LIST MAY HAVE REPEATS!

Short Version of Find Largest

1.) $j \leftarrow 1$

2.) for $i \leftarrow 2$ to n

3.) if $a_i > a_j$

4.) $j \leftarrow i$

5.) Print a_j, j

6.) stop

Problem: PATTERN MATCHING

GIVEN TEXT

$$T_1 T_2 T_3 \ldots\ldots T_n$$

FIND ALL OCCURRENCES OF A PATTERN

$$P_1 P_2 \ldots P_m$$

WHERE $m \leq n$.

Ex. $n = 18$, $m = 2$

text : 'to be or not to be'

T_1 T_2 T_3 ... T_{18}

Pattern: 'be'

P_1 P_2

Answer: 4, 17

Ex $n = 10$, $m = 3$

Text : ' x x x a a a a a a x x

T_1 T_4 T_{10}

Pattern : ' a a a '

P_1 P_2 P_3

ANSWER : 4, 5, 6

Ex. $n = 7, m = 3$

CHECK POSITION

T_1 T_2 T_3 T_4 T_5 T_6 T_7 — $i = 1$

P_1 P_2 P_3 ⟶ SHOE OVER

T_1 T_2 T_3 T_4 T_5 T_6 T_7 — $i = 2$

P_1 P_2 P_3 ⟶

T_1 T_2 T_3 T_4 T_5 T_6 T_7 — $i = 3$

P_1 P_2 P_3 ⟶

T_1 T_2 T_3 T_4 T_5 T_6 T_7 — $i = 4$

P_1 P_2 P_3 ⟶

T_1 T_2 T_3 T_4 T_5 T_6 T_7 — $i = 5$

P_1 P_2 P_3

RANGE OF POSSIBLE ANSWERS.

In general, range of possible answers is $\boxed{1}$ to $\boxed{n-m+1}$

Input: n, m satisfying $1 \le m \le n$.

text: $T_1 \ldots T_n$ to be searched

Pattern: $P_1 \ldots P_m$ to search for

Output: All indices i such that $P_1 \ldots P_m$ matches $T_i \ldots T_{i+m-1}$

PATTERN MATCH

```
1.) i ← 1
2.) while i ≤ n-m+1
3.)     j ← 1
4.)     match ← true
5.)     while j ≤ m and match
6.)         if P_j ≠ T_{i+j-1}
7.)             match ← false
8.)         else
9.)             j ← j+1
10.)    if match
11.)        Print 'match found at position' i
12.)    i ← i+1
13.) stop
```

Exercise: Insert this one the 'to be or not to be' and 'xxxaaaaaxx' examples.

Ex. 'hand the band to andy and I'

Space for 'and'

" " '□and□'

CHAPTER 3 : ATTRIBUTES OF ALGORITHMS

- CORRECTNESS
- CLARITY
- ELEGANCE
- EFFICIENCY

SEQUENTIAL SEARCH :

10.) if not found ←

OR 10.) if found = false

OR 10.) if $i > n$

ELEGANCE is ESSENTIALLY BREVITY:

Ex. 1.) get n
2.) SUM ← 0
3.) i ← 1
4.) while i ≤ n
5.) [SUM ← SUM + i
6.) [i ← i + 1
7.) Print SUM
8.) stop

1.) get n
2.) Print $\frac{n(n+1)}{2}$
3.) stop

OUTPUT is : $S = 1 + 2 + 3 + \cdots + (n-2) + (n-1) + n$

$$= \sum_{i=1}^{n} i$$

Formula: $S = \frac{n(n+1)}{2}$

Proof: $S = 1 + 2 + 3 + \cdots + (n-2) + (n-1) + n$

$S = n + (n-1) + (n-2) + \cdots + 3 + 2 + 1$

$S + S = (n+1) + (n+1) + (n+1) + \cdots + (n+1) + (n+1) + (n+1)$

$2S = n(n+1)$

$\therefore \boxed{S = \frac{n(n+1)}{2}}$