Algorithms and their cost

Algorithms- Set of steps for solving a problem

Input>Output

Sort- Put some sequence in an order

A = <54, 34, 23> input

A = <23, 34, 54> output

How do we evaluate an algorithm?

Assume: Algorithm is correct

Cost: numbers of instructions. For different input sizes and values

Algorithms complexity: General relationship between the input and the instructions.

Example 1: Write an algorithm to determine the index of specified value in an array (in pseudo code)

**findItem(A,v) //A is the array & v is the value**

**index = -1**

**for i = 1 to A.length**

**if A[i] == v**

**index = i**

**return index**

Pseudo code conventions:

* Not concerned about types
* Index starts at 1
* Assuming variable called A.length

Assuming this algorithm, what is the cost of A = <45, 34, 32, 3 4>?

**findItem(A,34)**

**findItem(A,v) //A is the array & v is the value**

**index = -1(1)**

**for i = 1 to A.length (4)**

**if A[i] == v (4)**

**index = i (2)**

**return index(1)**

Total: 12

Assuming this algorithm, what is the cost of A = <45, 34, 32, 34, 56, 23, 12 >?

Complexity:18

**findItem(A,34)**

**findItem(A,v) //A is the array & v is the value**

**index = -1(1)**

**for i = 1 to A.length (7)**

**if A[i] == v (7)**

**index = i (2)**

**return index(1)**

What input pattern of A will be worst case (most expensive)?

When all the items in the array is what you are looking for.

**findItem(A,v)**

**index = -1**

**for i = 1 to A.length**

**if A[i] == v**

**return i**

**return index**

Assuming this algorithm, what is the cost of A = <45,34,32,34>?

**findItem(A,v)**

**index = -1 (1)**

**for i = 1 to A.length (2)**

**if A[i] == v (2)**

**return I (1)**

**return index**

What if the worst case input? Item not found.

Example 2: Write an algorithm that finds a value in an array, removes it by shifting the array to fill empty spot

<54, 34, 45, 23, 90, 91, 0> remove 45

<54, 34, 45, 23, 90, 91, 0>

**removeItem(A,v)**

**for i = 1 to A.length**

**if A[i] == v**

**for j = i to A.length-1**

**A[i] = A[A+1]**

**A[length] = 0**

**return A**

A = <45, 34, 32, 36>

removeItem <A,34>

Cost = 16

**removeItem(A,v)**

**for i = 1 to A.length (4)**

**if A[i] == v (4, j=2-j=3)**

**for j = i to A.length-1 (2)**

**A[i] = A[A+1] (2)**

**A[length] = 0(1)**

**return A(1)**