



### Use of an array initializer

```

public class D {
    public static final String[] months= new String[]{"January", "February",
        "March", "April", "May", "June", "July", "August",
        "September", "October", "November", "December"};

    /** = the month, given its number m
        Precondition: 1 <= m <= 12 */
    public static String theMonth(int m) {
        return months[m-1];
    }
}
    
```

Months[m-1] is returned, since  
months[0] = "January",  
months[1] = "February",  
...

Variable months is:  
**static**: object assigned to it will be created only once.  
**Public**: can be seen outside class D.  
**final**: it cannot be changed.

7

### Procedure swap

```

public class D {
    /** = Swap x and y */
    public static void swap (int x, int y) {
        int temp= x;
        x= y;
        y= temp;
    }
}
    
```

A call will NOT swap a and b. Parameters x and y are initialized to the values of a and b, and thereafter, there is no way to change a and b.

....

a	5
b	3

swap:	1	?
-------	---	---

frame for call just after frame created and args assigned to pars:

x	5	y	3
temp	?		

8

### Procedure swap

```

public class D {
    /** = Swap b[h] and b[k] */
    public static void swap (int[] b, int h, int k) {
        int temp= b[h];
        b[h]= b[k];
        b[k]= temp;
    }
}
    
```

Does swap b[h] and b[k], because parameter b contains name of the array.

....

c	a0
---	----

a0
5
4
7
6
5

frame for call just after frame is created.

swap:	1	?	
b	a0	h	3
temp	?	k	4

9

### Linear search

```

public class D {
    /** = index of first occurrence of c in b[h..k-1] —
        = k if c is not in b[h..k-1] */
    public static int findFirst (int c, int[] b, int h, int k) {
        // if c is in b[h..k-1], return its index in b[h..k-1]
        // { invariant: c is not in b[h..t-1] }
        for (int t= h; t < k; t= t+1) {
            // Process t;
            if (b[t] == c)
                return t;
        }
        // { c is not in b[h..k-1] }
        return k;
    }
}
    
```

Remember  
h..h-1 is the empty range

10

### Two-dimensional arrays

	0	1	2	3
b	5	4	7	3

b.length    one-dimensional array

	0	1	2	3	
d	0	5	4	7	3
	1	4	8	9	7
	2	5	1	2	3

rectangular array: 5 rows and 4 columns

Type of d is **int[][]** ("int array array", "an array of int arrays")

To declare variable d:    number of rows  
**int d[][];**

To create a new array and assign it to d:  
**d= new int[3][4];**

To reference element at row r column c:    number of cols  
**d[r][c]**

11