CS125 – Craig Zilles Lecture 13 "Array Processing II" 9/25/2013

Objectives: Create and use 1D arrays. 2D arrays introduction.
Extra regrade of MP2Hollywood: Fri and Mon @8pm (99,98%)
Deadlines: Challenge 3 “Top Secret” due Monday.
Conflict Exam? Email lourent2@illinois.edu (incl. verifiable reason)

#1 Why is the following algorithm called selection sort?
(The method "findIndexOfMinimum" starts the search from index 'i' not index 0)

```
for(int i = 0 ; i < array.length ; i ++) {
    int smallest = findIndexOfMinimum(array, i);
    swap(array, i, smallest);
}
```

#2 Explain why the following do not make copies:

```
String s1 = "Hello!"
String s2 = s1;

int[] A = new int[] {101, 102, 103};
int[] B = A;
```

And explain why the following do not compare the values of the array or string objects:

```
// code continues from above
s2 = "Hello" + "!";
B = new int[] {101, 102, 103};

if ((A == B) && (s1 == s2)) TextIO.putln("Same!");
```

#3 What will be the final contents of 'myarray'?

```
String mesg = "Vewol Swap";
char[] myarray = mesg.toCharArray();

for (int i = 0; i < myarray.length; i ++) {
  if (myarray[i] == 'o') myarray[i] = 'e';
  if (myarray[i] == 'e') myarray[i] = 'o';
}
```

#4 You have a list of favorite movies, String[] movies
Write code to ask for another movie and append it to the end:

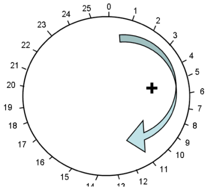
#5 Complete the following method to return the **array index** of the smallest value. Do not print anything out.

```
int findIndexOfMinimum(int[] array) {
   int smallest = 0; // index of smallest
   for(int i = 0; i < array.length; i ++) {

     if (array[i] _____________)smallest = ________

    }
   ______________
}
```

#6 MP3 Hints: "Modulo 26" ; negative integers may bite you.
29 % 26 is 3 (hurrah)
but
-3 % 26 is -3 not 23.



#7 Two dimensional arrays (MP3)

```
public char[][] make(int h) {
char[][] A= new char[h][h];
```

#6 What are the values of the array after the
// following code completes?
// y = row, x = column, assume h = 5

```
for (int y = 0 ; y < h ; y ++) {
  for (int x = 0 ; x< h ; x ++) {
    if ((x + y) == h) {
       A[y][x] = (char)('0' + x%2);
    } else {
       A[y][x] = ' ';
    }
    A[4-y][0] = '?';
  }
}
return A;
```

#8 Add just one more loop to change all of the outer border cells to be '*'

Col Row	0	1	2	3	4
0					
1					
2					
3					
4					

Notes

#9 // Returns true iff at least half the entries are positive.

```
public static boolean positive(double[] data) {



   for (int i = 0; i < data.length; i ++) {



  }


}
```

#10
// Returns true if there are at least 6 examples where the next array cell is twice the value as the previous one.
e.g. count ({1, 2, 4, 8 , 9, 3, 6, 0, 0, -1, -2 }) will return true.

```
public static boolean count(int[] data) {
  int result = 0;

  for (int i = 0; i < ____________________; i = i + 1) {

       if (_________________________)
                          result = result +1;



// don't forget the return statement
```