CS 125 - Lecture 13

**Objectives:**

Objectives: for-loops; Unicode representation; Introduce arrays;
Take your time reading arrays, they are important!  MP3 - have at it!

The following activities reinforce the readings....

1. **For the following code,**

```
for( int i=100; i>0 ; i = i / 10) {  TextIO.put(i); }
```

a. What does it print?

b. How many times is is i=i/10 evaluated?

c. How many times is i>0 evaluated?

d. Convert the above code into an equivalent while loop.

2. **Which examples will have the same behavior?**

```
//Read an integer value from the user :
int b=TextIO.getln();

// Followed by one of the following -
A) int  i; for(i=b; i<10 ; i++) { i= i*2;}

B) int i=b; for( ; i < 10; i++) i = i*2;

C) for(int i=b; i<10; ) {i=i*2;i++;}

D) for(int i=b; i<10; ); {i=i*2;i++;}

E) int i=b; while (i<10) {i=i*2;i ++;}

F) int i=b; while (i<10); {i=i*2;i ++;}

G) int i=b; do {i=i*2; i++ ;} while(i<10);
```

3. **What is the final value of i?**

```
int i=4; for(i-- ; i < 15; i++) { i = i * 2;}
```

4. **Convert the following code to use a for-loop:**

```
int count = 0;
int x = 7;
while(x < 50) {
  x = x * 2;
  count ++;
}
TextIO.putln("Final value:"+x);
```

5. **When do i and j go out of scope?**
**What does the following code snippet print?**

```
public static void main(String[]) {
   int i=4;
      while(i<6) {
         int j=1;
         while(j<3) {
            TextIO.put("("+ i + "," +j+")");
            if(j>1) TextIO.put(",");
            j++;
         }
         TextIO.putln();
         i++;
     }
}
```

6. **Write a program to print out all possible 2 letter words aa to zz:**
Hint use 2 for-loops.

7. **Write code to print the first 1000 unicode characters in a nicely formatted table.**

8. **The bank account swindler:**

```
double[] cash = new double[]
{-33,102,515,10004,42.07,…};
for(int i=0; i<cash.length; i++) {
   if(cash[myAccnt] < cash[i] ) {
      //Please Fix - It should swap the values.




      cash[myAccnt] = cash[i];









   } //if
} //for
```

9. **True/False?**

Q0. A Java array is an object. That is, a declared array variable does not actually hold the array, it refers to the array instead.

Q1. A Java array can hold a mixture of primitive types (e.g., integer in cell 0, boolean in cell 1, double value in cell 2, etc.)

Q2. The cells (or 'entries' or 'elements') of an array are indexed by an integer.

Q3. The first cell of an array is at index 1. To add 10 to the first score `scores[1] += 10;`

Q4. `scores.length = 500;` changes the size of the array.

Q5. The last cell in the array 'scores' will be `scores[scores.length -1]`

Q6. `new int[] {3,5,6,10}` creates an integer array of length 4.

Q7. `new int[99999]` creates a large integer array, each cell is initialized to zero.

Q8. `new char[50]` creates a character array with 50 cells, each cell is initialized to a space.

Q9. `new String[50]` creates a String array with each cell initialized to an empty string.

Q10. `scores[-1]` or `scores[scores.length]` will produce IndexOutOfBoundsException.

Participants in an experiment by Roediger and Karpicke (2006) either: 1) studied a passage four times, or 2) studied it once and then took three free recall tests. No feedback was given after each test. The total amount of time given to each group was the same.

Which group learned more?

3. **Does `scores[0]` change in the following code? Why?**

```
int[] scores = readScores();
String name[]= readNames();
int[] b = scores;
b[0] = 0;

TextIO.putln( name[0] + " : " + scores[0]);
```

4. **The bank account swindler:**

```
double[] cash = new double[] {-53,102,515,10004,42.07,...};
for(int i=0; i<cash.length; i++) {
   if(cash[myAccnt] < cash[i] ) {
      //Please Fix - It should swap the values.


      cash[myAccnt] = cash[i];


   } //if
} //for
```

6. **What will the following code return?**

```
public static String decode() {
     String secret = "Zfxrp";
     char[] mychars = secret.toCharArray();
     String result = "";
     for(int i=mychars.length - 1; i>=1;i--) {
        char c= mychars[i];
        result += (char) (c - 1);
     }
     return result;
}
// a b c d e f g h i j k l m n o p q r s t u v w x y z
```

7. **What will be the final contents of the array?**

```
int [] numbers = new int[] {10,11,12,13};

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

   numbers[i] = numbers[ numbers.length -1 - i];
```

Professor Jack Good, cryptanalyst working at the time with Turing at Bletchley Park, later said: "Turing's most important contribution, I think, was of part of the design of the bombe, the cryptanalytic machine. He had the idea that you could use, in effect, a theorem in logic which sounds to the untrained ear rather absurd, namely that from a contradiction you can deduce everything." (Source: Wikipedia)

The bombe searched for possibly correct settings used for an Enigma message (i.e., rotor order, rotor settings, etc.), and used a suitable "crib": a fragment of probable plaintext. For each possible setting of the rotors (which had of the order of $10^{19}$ states, or $10^{22}$ for the U-boat Enigmas which eventually had four rotors, compared with the usual Enigma variant's three), the bombe performed a chain of logical deductions based on the crib, implemented electrically. The bombe detected when a contradiction had occurred, and ruled out that setting, moving onto the next. Most of the possible settings would cause contradictions and be discarded, leaving only a few to be investigated in detail. Turing's bombe was first installed on 18 March 1940.

8. **Solving "Knight and Knaves" Logic Problems Computer Science Style!**

- Person 1 says "Person 2 is lying"
- Person 2 says "There are two liars here"

```
// 0 = liar, 1 = tells the truth
for (int person1 = 0; person1 < 2; person1++)
   for (int person2 = 0; person2 < 2; person2++){
      // Person 1 says "Person 2 is lying"
      boolean testimony1IsTruthful = person2 == 0;
      // Person 2: "There are two liars here"
      boolean testimony2IsTruthful = person1 + person2  == 0;

      boolean assertion1 = (person1 == 1 && testimony1IsTruthful)
                        || (person1 == 0 && !testimony1IsTruthful);
      boolean assertion2 = (person2 == 1 && testimony2IsTruthful)
                        || (person2 == 0 && !testimony2IsTruthful);

      TextIO.put("Person 1 is "
                 + (person1 == 0 ? "a liar" : "truthful")
                 + ", Person 2 is "
```

5. **Carefully execute the following code by hand and note the variables values as they change.** (i) Determine the final value of each variable. (ii) Determine what the code does.

| i: | j: | count: |
|---|---|---|
| result: | | |

```
int[] arr1 = {10, 20, 30, 40}; //sorted values
int[] arr2 = {18, 20, 25, 99}; //sorted values
int[] result = new int[arr1.length];
int i=0,j=0,count=0;
while (i<arr1.length) {
   if(arr2[j] < arr1[i]) j++;
   else if(arr2[j] == arr1[i]) i++;
   else { // must be true that arr2[j] > arr[i]
      result[count] = arr1[i];
      i++; count++;
   }
}
```

8. **PARALLEL ARRAYS:** Complete this code to print up to 50 movie titles of movies that grossed over $5 million. Print the array index of the highest grossing movie.

```
public static void main(String[] args) {
double[] gross = ... //gross[i] movie earnings of ith movie (in $m)
String[] title = ... //title[i] movie title of ith movie.
```