CS 125 - Lecture 17

Objectives: Multidimensional arrays; Hexadecimal and color representation;

Up next: Pair programming; MP4 due in 10 days;

1. Warm up … what does the following print?

```
for(int a=5; a>2; a--) {
   int b=a;
   while (b<2*a) {
      TextIO.put('*');
      b++;
   }
}
```

2. 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
```

3. Thinking in Base 16:

one hexadecimal digit = 4 bits.

Hex representation in java literal: 0x______

Convert 0xFF to binary.

Convert 0xFF3333 to binary:

Convert 0xBAADF00D to binary:

Convert $111111110001000001010101_2$ to hexadecimal:

10	Hex	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	
9	9	
10	A	
11	B	
12	C	
13		
14		
15		

Bit-wise operators: (& , | , ~ , ^)

a = 60_{10} = ____________________$_2$

b = 13_{10} = ____________________$_2$

a & b = ____________________$_2$ = ________$_{10}$

a | b = ____________________$_2$ = ________$_{10}$

a ^ b = ____________________$_2$ = ________$_{10}$

~b = ____________________$_2$ = ________$_{10}$

Bit-wise shifts:

Evaluate a << 2 ? ____________________$_2$ = ________$_{10}$

Evaluate a >> 2 ? ____________________$_2$ = ________$_{10}$

Evaluate 0x33ff33 >> 8 ?

4. Representing colors?

What color is (red=100%, green = 0%, blue = 0%)?

Design a solution to work in base 10. How would you represent this color as a single integer? e.g. This color is represented by the integer 900.

Now think in base2, What color is $111111111111111111111111_2$

CS 125 - Lecture 17

5. **What is this doing? What are the mystery variables representing?**

```
int mystery1 = (rgb) & 0xff;       // mystery1 is:

int mystery2 = (rgb >> 8) & 0xff;  // mystery2 is:

int mystery3 = (rgb >> 16) & 0xff; // mystery3 is:
```

Can you go in reverse?

```
int rgb = (_________<< 16) | (__________ << 8) | (__________);
```

6a. Fix / Complete the following code to initialize and return a square array of size h x h to a checker patter of "O" and "E" (O for 'odd' squares, E for even including [0][0]).

```
public static ____________ makeChecker(int h) {
   ______________ result = new _______________
   int i=0, j =0;
   for( ; i < result.length; i++ ) {
      for( ; j < result.length; j++) {
         if((i+j) ______________ )
            result________________
         else
            result _______________
      }
   }

}
```

6b. How should we test makeChecker?

7. **Using 2D arrays to represent an image.**

Create a picture of the JVMs memory and use memory pointers to explain why the following code swaps two rows.

```
int[][] pixels;
pixels = new int[480 /*row or 'y' coordinate*/][640 /* column or 'x'];
//  initialize pixel array : Odd rows are black.
//  Even rows are white
for(int y=0;y< 480; y++)
   for(int x = 0; x< 640; x++)
      if(y % 2 ==0) pixels[___][____] = 0xffffff;

//0xfffff = all white (red=255,green=255,blue=255)
int[] temp = pixels[10];
pixels[10] = pixels[11];
pixels[11] = temp;
```

CS 125 - Lecture 17

Objectives: images; 2-D arrays

7. What do the following do? Fix any syntax errors you notice.

```
new int[6];
new int[6] { 1,2,3,4,5,6 };
int[] a = {1,3,5,7,9,11};
int[] b=null;
b=a;
char[100] myvariable = new char[100];
int len = myvariable.length();
```

8. Why does the following cipher attempt fail for long messages? Can you fix it?

```
   String mesg = TextIO.getln();
   mesg = mesg.toUpperCase();
   String result = “";
   for(int x= 0; x< mesg.length(); x++) {
      char c = mesg.charAt(x);
      char encoded =(char)( 'A' + (c -'A' + x) );
      result += encoded;
   }
```

4. Why does the following cipher attempt fail for long messages? Can you fix it?

```
   String mesg = TextIO.getln();
   mesg = mesg.toUpperCase();
   String result = “";
   for(int x= 0; x< mesg.length(); x++) {
      char c = mesg.charAt(x);
      char encoded =(char)( 'A' + (c -'A' + x) );
      result += encoded;
   }
```

10. What does the following print?

```
   for(int a=5; a>2; a--) {
      int b=a;
      while (b<2*a) {
         TextIO.put('*');
         b++;
      }
   }
```

3. Merge...

Complete the following code to merge two sorted integer arrays together into a single output array

.

```
public static int[] merge(int[] A, int[] B) {
   int done = 0;
   int countA = 0;
   int countB = 0;
   int[] result = new int[______________________];
   while ((countA < A.length) ____ ___________________) {
      if (______________]) result[done++] = A[ countA++];
      else
       result[______] = B[ __________];
   }
   while (countA < A.length)
   result[done++] = A[countA++];
```

3. Love...

Complete the following code to print out a random love letter. Choose a random phrase from each string array.

```
public static void main(String[] args) {}
      String [][] letter = {
            {"Hi", “Dear","Dearest"},
            {"Mike,","Jenny,","sugar,","sweetheart,"},
            {“\n"},
            {"I can no longer","I want to","I need to”},
            {"think","swim","break up","sing country music"},
            {"for you.","with you.","about you.”},
            {“\n"},
            {"Bye,","Your loving friend,”},
            {“\n”},
            {“Jenny","Jim"}
      };
```

1. Fix / Complete the following code to initialize and return a square array of size h x h to a checker patter of "O" and "E" (O for 'odd' squares, E for even including [0][0]).

```
public static ____________ makeChecker(int h) {
   _______________ result = new ________________
   int i=0, j =0;
   for( ; i < result.length; i++ ) {
      for( ; j < result.length; j++) {
         if((i+j) ______________ )
            result_________________
         else
            result ________________
      }
   }
```

1b. How should we test makeChecker?

3. Complete the following bucket sort code to sort the data array.

```
int[] data = {5,22,5,18,4,…. 74623 more values between 0 & 999

int max = 1000;
int[] histogram = new int[max];

// Phase 1, count the number of occurrences of 0,1,2,3... max-1

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

   ? ________________________________________


// Phase 2, Use histogram to create the sorted output data

int ptr=0; // we will write values into data[ptr]

for(int value=0; value<max; value ++)

?

// This sort is fast but what limitations can you see with this
algorithm?
```