

PRACTICE FOR QUIZ 3

CMPS 12a - Spring 02
Thomas Raffill

Name: _____
Student ID: _____

This exam is closed book, closed notes, no electronic devices. Show all work. Partial credit given for partial solutions. Presentation counts! Be legible and coherent for full credit.

Question 1: _____ (out of 12)
Question 2: _____ (out of 16)
Question 3: _____ (out of 12)
Question 4: _____ (out of 12)
Question 5: _____ (out of 12 + 6 extra credit)
Question 6: _____ (out of 12)
Question 7: _____ (out of 12)
Question 8: _____ (out of 12)
Question 9: _____ (out of 12 extra credit)

Total: _____ (out of 118)
(Anything above 100 counts for extra credit)

Name: _____

1. (12 points)

You have an array called `gradeList` containing `int` values. You want to modify each value in the array by multiplying it by 10. Write a code fragment to step through the array in a loop and accomplish this.

Solution: Here is one way to do it.

```
for (int i = 0; i < gradeList.length; i++)
{
    gradeList[i] *= 10;
}
```

2. (16 points)

You would like to create a `String` array named `daysOfWeek` containing `String` values "Mon", "Tue", "Wed", "Thu", "Fri", "Sat" and "Sun". Show two different ways to create the array and fill it with these value.

Solution:

Method 1: `String[] daysOfWeek = { "Mon", "Tue", "Wed", "Thu", "Fri", "Sat", "Sun" };`

Method 2: `String[] daysOfWeek = new String[7];`
`daysOfWeek[0] = "Mon";`
`daysOfWeek[1] = "Tue";`
`daysOfWeek[2] = "wed";`
`daysOfWeek[3] = "Thu";`
`daysOfWeek[4] = "Fri";`
`daysOfWeek[5] = "Sat";`
`daysOfWeek[6] = "Sun";`

Name: _____

3. (12 points)

Show how you can use an array to represent your scores on the nine homework assignments in this class. You should give a small code fragment and verbal explanation.

Solution:

```
int[] homeworkScores = new int[9];
```

This code creates an array of nine int values to contain the nine homework scores.

Name: _____

4. (12 points)
What does the following program print out?

```
public class QuizArrays
{
    public static void main(String[] argv)
    {
        int[] quizArray1 = new int[10];
        int[] quizArray2 = new int[10];
        quizArray2 = trickyArrays(quizArray1, quizArray2);
        System.out.println("The first element of quizArray1 is " +
                            quizArray1[0]);
        System.out.println("The first element of quizArray2 is " +
                            quizArray2[0]);
    }

    public static int[] trickyArrays(int[] firstArray, int[] secondArray)
    {
        firstArray[0] = 111;
        secondArray[0] = 222;
        return firstArray;
    }
}
```

Solution:

It prints out:

The first element of quizArray1 is 111

The first element of quizArray2 is 111

5. (12 points)
Suppose you have a 2-dimensional array of int values called
int twoDQuizInts[], and you have int variables called numRows and num-
Cols containing the number of rows and columns in the array. Write a
code fragment to step through the array in a double loop, find the mini-
mum element in the array, and print out the result.

Solution:

```
int min = twoDQuizInts[0][0];
for (int r = 0; r < numRows; r++)
{
    for (int c = 0; c < numCols; c++)
```

```

        {
            if (twoDQuizInts[r][c] < min)
        {
min = twoDQuizInts[r][c];
}
        }
    }
    System.out.println(min);

```

Extra Credit:(6 points)

Do the above, but don't use the numRows and numCols variables to control the loops.

```

int min = twoDQuizInts[0][0];
for (int r = 0; r < twoDQuizInts.length; r++)
{
    for (int c = 0; c < twoDQuizInts[r].length; c++)
    {
        if (twoDQuizInts[r][c] < min)
        {
            min = twoDQuizInts[r][c];
        }
    }
}
System.out.println(min);

```

6. (12 points)

Java has a class called StringBuffer. You can create a StringBuffer variable as follows:

```
StringBuffer b = new StringBuffer(s)
```

and the class has an instance method called reverse() which reverses the order of the letters in the StringBuffer. Write a code fragment which creates a StringBuffer for "abracadabra" and reverses the letters to "arbadacarba".

Solution:

```
StringBuffer b = new StringBuffer("abracadabra");
b.reverse();
```

Name: _____

7. (12 points)

Consider the following piece of code:

```
public static void trickySwitch(int x)
{
    switch(x % 3)
    {
        default: System.out.println("Is this line ever printed?");
        case 0:
            System.out.println("The number is divisible by 3.");
        case 1:
            System.out.println("The number is 1 mod 3.");
            break;
        case 2:
            System.out.println("The number is 2 mod 3.");
    }
}
```

(a) What is printed out if the value of x passed in is 5?

Solution: The number is 2 mod 3.

(b) What is printed out if the value of x passed in is 6? **Solution:**

The number is divisible by 3.

The number is 1 mod 3.

(c) What is printed out if the value of x passed in is 7? **Solution:** The number is 1 mod 3.

8. (12 points)

Here is a code fragment with a potential `ArrayIndexOutOfBoundsException`. Place the code within try-catch blocks so that if the array index is out of bounds, your code will print out the message, "The loop goes too far!"

```
public static void riskyLoopFunction(int x, String[] s)
{
    for (int i = 0; i < x; i++)
    {
        System.out.println(s[i])
    }
}
```

Solution:

```
public static void riskyLoopFunction(int x, String[] s)
{
    try
    {
        for (int i = 0; i < x; i++)
        {
            System.out.println(s[i]);
        }
    }
    catch(ArrayIndexOutOfBoundsException e)
    {
        System.out.println("The loop goes too far!");
    }
}
```

Name: _____

9. **Extra Credit:**(12 points)

Here is a code fragment with a potential `IOException`, a potential `NumberFormatException` (parsing a `String` which might not be a number), and a potential `ArithmeticException` (division by 0). Place the piece of code in a try block and add three catch blocks to catch these three different kinds of Exceptions. If it is an `IOException`, your code should print out the message, "Trouble reading file." If it is a `NumberFormatException`, your code should print out the message, "The file content is not a number." If it is an `ArithmeticException`, your code should print out the message, "Problem with division by 0."

```
BufferedReader quizInput = new BufferedReader(new FileReader("quizFile.txt"));
String quizInputLine = quizInput.readLine();
int x = Integer.parseInt(quizInputLine);
double y = (double)10/x;
System.out.println("The answer is " + y);
```

Solution:

```
try
{
    BufferedReader quizInput = new BufferedReader(new FileReader("quizFile.txt"));
    String quizInputLine = quizInput.readLine();
    int x = Integer.parseInt(quizInputLine);
    double y = (double)10/x;
    System.out.println("The answer is " + y);
}
catch(IOException e)
{
    System.out.println("Trouble reading file.");
}
catch(NumberFormatException e)
{
    System.out.println("The file content is not a number.");
}
catch(ArithmeticException e)
{
    System.out.println("Problem with division by 0.");
}
```