

lab 10

Program Design

(Solutions in lab10_Hangman and lab10_RockPaperScissors)

1. Reference the week10_design.pdf file in the week10 folder on Brightspace. Finish the prompts at the end of the Rock, Paper, Scissors game we completed in class.
 - Can an array be added anywhere in the game? It doesn't have to but investigate if it can.
 - Can the counting wins logic be moved to a separate method? Make the `main` method as simple as possible.
2. Complete the Hangman game in the week10_design.pdf file. The game should randomly select a word from an array of words, and then prompt the user to guess the word. The user should have 6 attempts to guess the word. The game should display the word with asterisks for the letters that have not been guessed. The game should display the word and a message if the user guesses the word or runs out of attempts.
 - I have provided comments that outline the different parts of the game. Try your best! If you can't get it to work that's okay. We will go over the solution in class. Don't use the whole lab time on this question. If you get stuck, move on to the next question.
 - You can also design your own solution that doesn't follow my comments. The important thing is that you are practicing your programming skills. Good luck!

Array Practice

3. Create a method that creates a copy of an int array. The method should take an int array as a parameter and return a new array that is a copy of the original array.
 - Hint: You need to create a new array of the same size as the original array. Then you can copy the values from the original array to the new array.

```
public static int[] copyArray(int[] array) {  
    int[] copy = new int[array.length];  
    for (int i = 0; i < array.length; i++) {  
        copy[i] = array[i];  
    }  
    return copy;  
}
```

4. ☐ grades = {98.6, 78.2, 56.9}; Suppose these represent test grades for a student. The professor decides to give an extra exam. Write code to create a larger array to hold the old grades plus the new exam grade of 88.8.

- Hint: You need to create an array one size larger than the original array. `array.length` will give you the size of the original array.
- Then you can add the new grade to the end of the new array.

```
double[] grades = {98.6, 78.2, 56.9};
double[] newGrades = new double[grades.length + 1];
for (int i = 0; i < grades.length; i++) {
    newGrades[i] = grades[i];
}
newGrades[grades.length] = 88.8;
```

5. ☐ `a={56, 9, 17, 6, 2, 0, 1, 199, 256, 94};`

- Create an array to hold the even values of `a`.
- Hint: First count how many even numbers there are so you know what size to make the new array.
- Create an array to hold the odd values of `a`.

```
int[] a = {56, 9, 17, 6, 2, 0, 1, 199, 256, 94};

int count = 0;
for (int i = 0; i < a.length; i++) {
    if (a[i] % 2 == 0) {
        count++;
    }
}

int[] even = new int[count];
int index = 0;
for (int i = 0; i < a.length; i++) {
    if (a[i] % 2 == 0) {
        even[index] = a[i];
        index++;
    }
}

//Just change the if statement to get the odd numbers
```