# AP Computer Science
# Unit 2 Test Review

**Online Resources**

1. On APClassroom:
    a. The Unit 2 Review quiz. These are questions I selected as a mixed review, and you should do this at least for multiple choice review.
    b. Units 2 and 4 progress check MCQs. These are good for additional multiple choice practice. (Units 2 and 4 in the APClassroom organization are both part of this unit.)
    c. Daily videos. Useful if you want a direct instruction review of a particular topic. There are too many to watch all of them, target particular things you want to review.
2. In CSAwesome:
    a. Sections 2.10, 2.13 and 2.15 for review of using objects and methods. (Ignore the parts about the Integer and Double classes, and wrapper classes. We haven't talked about those yet.)
    b. Sections 4.6, 4.10 and 4.11 for review of loops.

**Example Test Programming Questions**

I aim for the coding problem on the test to be reasonable to do in about 30 minutes. You will not be allowed to access outside resources, but you will be able to use the APCS Quick Reference Sheet (provided in class, posted on Google Classroom). Also, if you have questions about specific syntax (like "what's the line for creating the Scanner again?") I will answer them.

Example problem 1

The "Unit 2 Review" project on Replit has a `BankAccount` class available in it. All of the information you need in order to use the class is contained in the method signatures below. (So you don't actually need to look at the `BankAccount` class to complete the `BankAccountTester`.)

| Method | Description |
|---|---|
| `public BankAccount(String accountName)` | Constructs a `BankAccount` for a person with the given `accountName`. Initially the balance in the account is $0, and the account is given a random 8-digit account number. |
| `public String getName()` | Returns the name of the account holder. |
| `public int getAccountNumber()` | Returns the 8-digit account number for this account. |
| `public void deposit(double amount)` | Adds the specified amount of money to the account balance. |
| `public String toString()` | Returns a `String` with the name of the account holder and the current account balance. |

In the `BankAccountTester` class, write a program that does the following: create new bank accounts until you create one where the first digit of the account number is a 6. Then deposit an amount of money into that account equal to the account number, and print the account.

Example problem 2

Write a program that asks the user to enter a String, and then prints the first character of that String on the first line, the first two characters on the second line, etc., until it has printed the whole String. For the purposes of this problem, use the `charAt()` method, not the `substring()` method. (The purpose of this problem is to help you practice nested loops in the context of a problem that would be reasonable for a test, and using `substring()` lets you avoid that.)

For example, if the user enters “Wait”, your program would print:

```
W
Wa
Wai
Wait
```

Example problem 3

Write a program that asks the user to enter an integer. Then ask the user to enter a second integer that is larger than the first one. Your program should then calculate the sum of all of the multiples of 3 that are between the first and second numbers (inclusive) that were entered. For example, if the numbers 5 and 11 are entered, the program would print 15, because 6 and 9 are the two multiples of 3 between 5 and 11 and 6 + 9 = 15:

```
Enter an integer:
5
Enter an integer that is bigger than the previous one:
11
The sum of the multiples of 3 between 5 and 11 is: 15
```

If one (or both) of the numbers that were entered is a multiple of 3, that number would be included in the sum:

```
Enter an integer:
6
Enter an integer that is bigger than the previous one:
12
The sum of the multiples of 3 between 6 and 12 is: 27
```

(Note that 6 + 9 + 12 = 27)

**Other Programming Practice**

Write a program that prompts the user for a String, and then determines and displays the character that is repeated the largest number of times in a row. It should print that character, followed by the number of times in a row that it appears. For example, if the user enters “A CCAAAAATTT!", your program should print the result "A 5" because the longest substring of consecutive identical characters is "AAAAA".