Functions/Methods Practice Problems

# True/False Questions

1. Every program needs a MAIN function or method to run.
2. Methods can access and modify other methods’ local variables easily.
3. The bottom of the runtime stack is the currently running method.
4. The top of the runtime stack is the currently running method.
5. Methods can be called from non-MAIN methods.
6. One method can call another method
7. Only the main method can call another method.

# Multiple Choice Questions

1. How many times will this method1() be called with the following code?

for(int i=7;i<10;i++)

{

method1();

}

* 1. 3
	2. 7
	3. 10
	4. 4
1. Which of the following does not have any effect in method overloading?
	1. Number of parameters
	2. Return type
	3. Types of parameters
	4. Method name
2. Creating multiple methods with the same name but different parameters (number of parameters and/or data types) is called:
	1. Overriding
	2. Objections
	3. Overloading
	4. Overwriting
3. Which of the following is an advantage of creating and using methods in programs?
	1. Increase complexity of code
	2. Make program maintenance and updates harder
	3. Make code more obscure
	4. Allow reuse of code
4. Modularizing code by using functions or methods:
	1. Reduces redundant code
	2. Increases maintenance time
	3. Eliminates encapsulation
	4. Requires a return statement
5. A Method’s signature is comprised of:
	1. Method Name and Return type
	2. Method Name and Parameter List
	3. Return Type and Parameter List
	4. Method Header and Body
6. This means that a copy of a parameter’s value is sent into a method:
	1. Pass by reference
	2. Pass by value
	3. Passover
	4. Parameter list
7. Methods are used to:
	1. Allow code reuse
	2. Reduce code complexity and length
	3. Make program maintenance easier.
	4. All these reasons

# Programming & Problem-Solving Questions

1. Assume that the average person can consume 2500 calories in one day without gaining any weight. There are 3500 calories in one pound of fat. Your job is to write a function/method that takes in two parameters:
	1. the number of calories per day that a person eats
	2. the number of years a person will eat that many calories.
	3. The program will return the number of pounds the person will gain (and technically work for losing weight as well). Yes, there are 365 days in one year! Note: a person that eats 2500 calories a day will not gain weight (for this question). Why?
2. Write a method, called CheckLetter. The method receives a letter as a parameter and returns whether the letter is a lowercase vowel (a, e, i, o, u) or not. Sample outputs are:

The entered letter is: a
a is a vowel.

The entered letter is: b
b is not a vowel.

1. Write a method max (x, y, z) that determines and returns the maximum value of three integer values.
2. Write a function that accepts a father’s birth year and his child’s birth year as arguments then determines what year the father will be twice as old as the son and returns the calculated year to the main method.
3. Write a function that accepts a single number, calculates the number of positive odd numbers below the parameter value and returns the calculated result to the main method.
4. Write a function that determines if a user is playing the banjo by passing their name as a parameter, if their name starts with A through L they are not playing the banjo. Return a statement indicating if they are playing or not to the main method.
5. Write a function that accepts the subtotal from a meal, the tax rate and the tip percentage, it should calculate the total (do not tip on the tax) and return the total to the main method.
6. Write a function that accepts a student’s final exam grade (worth 40 points) and number of completed projects (each worth 10 points) and returns the calculated final grade (out of 100 points) to the main method.
7. Imagine that you are writing a secure login that determines whether or not a name+password combination is correct.  Since you’ve just started to test the system, the only name you will accept is “admin” with the password of “password”.  Write a function that takes in two strings and returns whether or not those strings are “admin” and “password”.  Note, if the name is “admin” and the password is not “password”, it has failed the criteria.
8. Imagine you’re writing a game where you need to determine if a player has gone off the left or right side of the screen – or “out of bounds”. Write a function that takes in 3 numbers (x1, x2, w) and returns whether or not the value of w is between that of x1 and x2.  You can assume that x1 is less than x2.
9. Imagine that a bank only pays interest of 1% (0.01) each month on an account if it has more that $1000 in it.  If the account has less than $1000, it should return 0. Write a function that takes in a dollar amount and returns the correct amount of interest.
10. Write a method that takes in an array of lowercase characters and returns the number of vowels (a, e, i, o, or u) in the array. C++ students: you can pass in an additional parameter that represents the array length
11. Write a method that verifies that an array of booleans (sent as a parameter) follows a pattern of true/false/true/false… The method should return a 1 if pattern matches or -1 if it does not**.** C++ students: you can pass in an additional parameter that represents the array length.
12. Write a method that compares two integer arrays (sent in as parameters) and returns true if all elements match. Otherwise, it should return false. C++ students: you can pass in an additional parameter that represents the array length.
13. Write a function named PowerXY() (only the function, no main is necessary) that takes in two numbers and returns the first number to the power of the second (e.g. passing 3 and 4 should return 81 because 3^4 = 81)
14. Write a function named TrueFalse() (only the function, no main is needed) that takes in a number and determines if it is evenly divisible by 5 (i.e. returns true or false).
15. Write a function (and only the function) that takes in a character and returns whether or not that character is a vowel.
16. Write a function (and only the function) that takes in two numbers and returns the sum of all the numbers between (and including) those two numbers
17. Write a function named MinXYZ (only the function, no need for main) that takes in three numbers and returns the minimum of those three numbers
18. Write a METHOD takes in two non-equal negative integers and returns the difference between them to the method call. The method should include two parameters (appropriately typed), perform the calculation and return the result.
19. Write a METHOD that takes in two positive non-equal numbers and returns their ratio (a/b) as a decimal. The method should include two parameters (appropriately typed), perform the calculation and return the result.
20. Write a METHOD that takes in two integers (one positive and one negative) and returns the sum of the two integers to the method call. The method should include two parameters (appropriately typed), perform the calculation and return the result.
21. Write a METHOD that takes in two non-equal integers and returns the product of the two numbers to the method call. The method should include two parameters (appropriately typed), perform the calculation and return the result.
22. Write a METHOD takes in two non-equal negative integers and returns the difference between them to the method call. The method should include two parameters (appropriately typed), perform the calculation and return the result.
23. What is the exact output of the following pseudocode segment?

METHOD MAIN

CALL myMethod (0,2)
CALL myMethod (3,5)

CALL myMethod (6,7)

END MAIN

METHOD myMethod(A,B)
BEGIN
 WHILE (A < B)
 PRINT(A + " ")
 A ← A + 1
 ENDWHILE
 PRINTLINE();
END myMethod

1. What is the exact output of the following pseudocode segment?

METHOD MAIN

CALL myMethod (2, 0)
CALL myMethod (5, 3)

CALL myMethod (7, 6)

END MAIN

METHOD myMethod(A,B)
BEGIN
 WHILE (A > B)
 PRINT(A + " ")
 A ← A - 1
 ENDWHILE
 PRINTLINE();
END myMethod

1. Fair Treats: Write a METHOD to print out this menu of topping choices for Candy Apples at a fair:
	1. Sprinkles
	2. Peanuts
	3. Chocolate Chips
	4. Write a METHOD that prompts the user to make their selection as an Integer and return the user’s selection from the menu as a String.
	5. Write the MAIN program that calls the Selection Method after calling the Print Menu method and then prints a statement verifying which topping they selected.
2. Homecoming: Write a METHOD to print out this menu of sports teams to display their number of winning seasons (seasons with more wins than losses):
	1. Berry College Vikings
	2. Reinhardt Eagles
	3. Kennesaw State University Owls
	4. Write a METHOD that prompts the user to make a selection as an **Character** and returns the user’s selected result from the menu as an **Integer**.
	5. Write the MAIN program that calls the Selection Method after calling the Print Menu method and then prints a statement showing how many winnings seasons the chosen team currently has.
3. Halloween Costumes:
	1. Write a METHOD to print out this menu of possible costume rentals for Halloween:
	2. Thanos
	3. Darth Vader
	4. Zombie
	5. Write a METHOD that prompts the user to make a selection as a Character and returns a Boolean indicating whether or not the user’s selection from the menu is available to rent.
	6. Write the MAIN program that calls the Selection Method after calling the Print Menu method and then prints a statement indicating whether that costume is available for rental.
4. Pumpkin Patch: Write a METHOD to print out this menu of possible pumpkin weights at the pumpkin patch:
	1. Small – up to 5.0 pounds
	2. Medium – up to 7.5 pounds
	3. Large – up to 9.99 pounds
	4. Write a METHOD that prompts the user to make a selection as a Double (weight of desired pumpkin) and returns the user’s pumpkin weight from the method as a Double.
	5. Write the MAIN program that calls the Selection Method after calling the Print Menu method and then prints a statement verifying which size pumpkin they selected based on its’ weight.
5. Pumpkin Carving: Write a METHOD to print out this menu of pumpkin carving designs:
	1. Triangular eyes and oval mouth
	2. Circular eyes and circular mouth
	3. Circular eyes and oval mouth
	4. Write a METHOD that prompts the user to make a selection as an Integer and returns the user’s selection from the menu as a String.
	5. Write the MAIN program that calls the Selection Method after calling the Print Menu method and then prints a statement verifying which design they have chosen for their pumpkin.
6. Prize Choices: Write a METHOD to print out this menu of choices for prizes at a fair:
	1. Stuffed Animal
	2. Flag
	3. Cape
	4. Write a METHOD that prompts the user to make a selection as an Integer and returns the user’s selection from the menu as a String.
	5. Write the MAIN program that calls the Selection Method after calling the Print Menu method and then prints a statement verifying which prize they have selected for winning a carnival game.
7. Flash Pass: Write a METHOD to print out this menu of Six Flag rides to get a Flash Pass for:
	1. BATMAN: The Ride
	2. Georgia Scorcher
	3. Justice League: Battle for Metropolis
	4. Write a METHOD that prompts the user to make a selection as an Integer and that returns a user’s selection from the menu as a Character
	5. Write the MAIN program that calls the Selection Method after calling the Print Menu method and then prints a statement verifying which ride they want the pass for.