**BCS 230 Lab – C String**

***Overview***

Write a program that uses C Strings.

***Part 1***

Write a program that will simulate c-string operations:

1. Declare three different char arrays.
2. Put char data in each array. The data should be null terminated.
3. Create a method called “print” that takes one char array as a parameter. This method should print the contents of the array on standard output. You should only print data that appears before the null terminator. You should print the data ONE character at a time.

* DO NOT CALL THE BUILT-IN STRING METHODS. YOU ARE WRITING THEM YOURSELVES!!!
* ONLY USE CHAR CONSTANTS. DO NOT USE STRING CONSTANTS ANYWHERE IN THIS PROGRAM!!!

MAKE SURE YOU ADD NULL TERMINATORS WHERE NECESSARY.

***Part 2***

Create a method called “copy” that takes two char array parameters. This method should copy the data in the second parameter into the first parameter. You should use a loop in this method.

***Part 3***

Create a method called “concat” that takes three char array parameters. This method should concatenate the second string on to the end of the first string and store the result in the third string.

Note: When you concatenate two strings together you add the contents of one string on to another.

Important: Make sure you add null terminators where necessary.

***Part 4***

Create a method called “compare “ that takes two char array parameters and returns an integer. This method should compare the first char array to the second char array. The return value should be calculated as follows:

-1 if first parm < second parm

0 if first parm == second parm

1 if first parm > second parm

***Part 5***

Create a method called “find” that takes two char array parameters and returns an integer. This method should search the first char array for the data in the second char array. Basically, does the string in the second char array exist in the first char array. The return value should be calculated as follows:

If the second string is contained in the first string then

return the index of where it was found in the first string.

If the second string is not found then return -1.