Lab 7

Each lab will begin with a recap of last lab and a brief demonstration by the TAs for the core concepts examined in this lab. As such, this document will not serve to tell you everything the TAs will in the demo. It is highly encouraged that you ask questions and take notes. In order to get credit for the lab, you need to be checked off by the end of lab. For non-zero labs, you can earn a maximum of 3 points for lab work completed outside of lab time, but you must finish the lab before the next lab. For extenuating circumstance, contact your lab TAs and Instructor.

(6 pts total) Design First (3 pts), Then Write Code (3 pts)

To help you practice functions, you will write a short program that asks the user to enter a string (**get_string**()), makes a copy of the string, takes the copy and changes all non-space letters to dashes (**set_replace_string**()). The program then gets a letter from the user to search in the original string and replace the dashes in the copy string with the letter found, returning the number of letters found (**get_search_replace**()).

You should design first. You can have more functions, but you must have at least the three below with the EXACT function prototypes.
Each of your functions should have the proper function headers/descriptions

```
void get_string(string *);
void set_replace_string(string, string *);
int get_search_replace(char, string, string &);
```

Write the function headers/descriptions for each of the functions above, as well as all the functions you create. This includes information about parameters, return values, and pre/post conditions.

Design – Give as much detail as possible for the main function and all the functions above.

- Why do the function prototypes have the specific parameter types on them?
- Why do the functions have void or a return value?
- How do all the functions interact together?

Testing – Provide testing values with expected results.

- What do you plan to use as bad values? Good values?
- What do you expect to happen with bad values? Good values?

Get checked off by a TA before beginning to implement. This will help with logic and function mistakes**.**

Now, incrementally write your functions and program.

(4 pts) Understand Pointers/Dynamic Memory
Write 3 different functions in C++ to create memory on the heap without causing a memory leak. Hint: You will need to assign the address to a pointer that was created outside the function. Remember, you can return an address or change what a pointer points to (the contents of a pointer) in a function by passing the pointer by reference or by passing the address of the pointer.

What if you want to change the contents of what the pointer points to? **Make a function that will set the contents of the space on the heap**. Do you still need to pass by reference or the address of the pointer? Why or why not?

How will you delete the memory off the heap? Try doing it outside a function, now inside a function. **Make sure your delete function is setting your pointer back to NULL**, since it is not supposed to be pointing anywhere anymore.

You can check to see if you have any memory leaks using valgrind.
%valgrind program_exe