***Exercise 5***

*Name Scope(HTML)*

1. Passing parameters by reference gives the function programmer the ability to "return" more than one value to the calling function, but passing a value *by reference*, which *does* allow the called function to change the value passed in the calling function. A call by reference looks exactly like a call by value *except* the “&” symbol is appended to the type in *both* the prototype and the function definition header
2. Automatic memory is allocated by the compiler within the total memory space allocated to a program (when it is loaded and run); this memory space includes memory allocated to variables and the program's executable code. For this reason, it is often called static memory (as opposed to dynamic memory).

Stack memory is dynamic memory, and is used when you make function calls. Stack memory usage changes in size: it increases when functions are called, and it decreases when functions return. A running program's stack memory is usually completely separated from its automatic memory; i.e., it occupies a different region within the total memory space.

double variable1; GLOBAL SCOPE

extern int variable2; GLOBAL SCOPE

void function1 (int&, double); PROTOTYPE

void function2 (int, double&); PROTOTYPE

int main () {

int variable1; LOCAL MAIN SCOPE

double variable2; LOCAL MAIN SCOPE

...

function1 (variable1, variable2); CALL FROM GLOBAL SCOPE

...

return 0;

}

void function1 (int& variable1, double variable2) {

static short variable3; LOCAL SCOPE AND ASSIGNED TO function1

...

variable3 = variable1 + variable2; LOCAL SCOPE

...

variable1 = variable3; LOCAL SCOPE

...

function2 (variable1, variable2); LOCAL SCOPE

...

return;

}

void function2 (int variable1, double& variable2) {

short variable3; LOCAL SCOPE AND ASSIGNED TO function2

...

variable2 = variable1 + variable3 LOCAL SCOPE

...

return;

}

*Function Design*

* Create a variable for first name, last name, and phone number.
* Prompt user for a 1 or 2.
* If the user input is 2, then the program searches by first and last name, but gives the phone number.
* Else if input 1, then the program searches by phone number, but gives the first and last name
* Search by names, the program opens a file, which loops in with regard to the phone number entered, and if the number is equal to the input, then output the first and last name.
* Else, nothing happens and the program terminates

1. //ignore, Microsoft won’t let me skip 2 before I get to number 3….. ugh…
2. In .cpp file