

Course	CSC 215 – Programming Techniques
Name	Homework 12
Due Date	April 3 at 12:00 (noon)
Repository	Place in homeworks repository
Solution Name	hw12.sln
Project Files	hw12Double.cpp

Download hw12.zip from d2l (<https://d2l.sdbor.edu>) and save this to your computer.

Double click the downloaded zip file.

Highlight the contents and Press <ctrl>-C or click copy from the home tab menu

Browse to your homework repository and paste the contents there.

<ctrl>-V

Right click in the contents window and select paste

Click copy from the home tab menu.

Change into this folder and open hw12.sln

You will be writing your code in the hw12Double.cpp. Hw12.cpp contains the tests for your code and the doubly linked list class functions

This project uses the 64bit version only. If you are getting message the x86 is not compatible with x64 you will need to change to 64bit in the configuration manager. (Build → Configuration Manager).

Functions: There are 7 functions that you must rewrite by Tuesday at noon. You must have it pushed to your repository.

- 1) isEmpty - returns true if the list contains no items, false otherwise
- 2) size - returns the number of items in the list
- 3) push_back - will add an item to the end of the list, returns true if successful
- 4) push_front - will add an item to the front of the list, return true if successful
- 5) pop_back - will remove an item at the back of the list, returns true plus the string value that was removed
- 6) pop_front - will remove an item at the front of the list, returns true plus the string value that was removed
- 7) swap - will exchange the contents of two lists.

Work on one function at a time. You will need to filter the tests being run by using visual studios command line options.

- 1) Enter "isEmpty"**
- 2) Enter "Size"**
- 3) Enter "Push Back", then Enter "Push Front", and Finally "Push"**
- 4) Enter "Pop Back", then Enter "Pop Front", and Finally "Pop"**
- 5) Enter "swap"**

The order of rewriting the functions can make your life easier. Do them in the order listed in functions.

Think about them. Do not drop to the command line to run the program.

You will need to delete the word "**UNSORTEDDOUBLE_API**" from in front of the function in the unsortedDouble.h file. Then write the function for this function. This will execute your function rather than the function within the DLL.

Example

The first function you should write is the constructor `unsortedDouble()`

In the header file, delete `UNSORTEDDOUBLE_API` from the constructor prototype.

Should look like the following

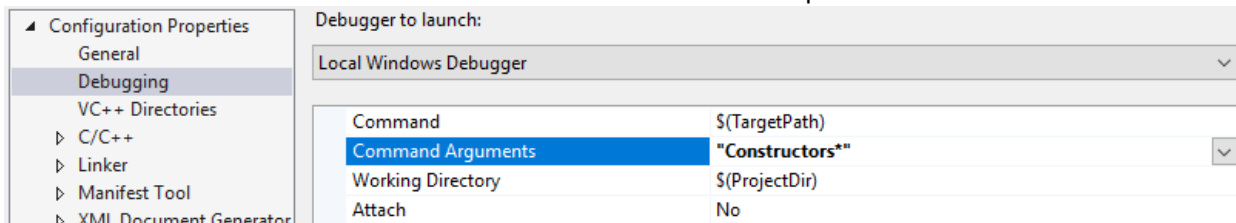
Before:

```
class unsortedDouble
{
public:
    UNSORTEDDOUBLE_API unsortedDouble ();
    UNSORTEDDOUBLE_API ~unsortedDouble ();
    UNSORTEDDOUBLE_API void clear ();
    UNSORTEDDOUBLE_API void print ( ostream &out, bool forward = true,
                                   string seperator = ", ");
    UNSORTEDDOUBLE_API bool insert ( string val, int pos=1);
```

After:

```
class unsortedDouble
{
public:
    unsortedDouble ();
    UNSORTEDDOUBLE_API ~unsortedDouble ();
    UNSORTEDDOUBLE_API void clear ();
    UNSORTEDDOUBLE_API void print ( ostream &out, bool forward = true,
                                   string seperator = ", ");
    UNSORTEDDOUBLE_API bool insert ( string val, int pos=1);
```

Place the correct filter for which tests to run in the command line options within visual studio.



Now write your code in the file hw12Double.cpp

```
#include "unsortedDouble.h"
#include <iostream>
#include <string>
#include <functional>

using namespace std;

// write your functions here.
```

You can download the documentation for each of the function from d2l.

Do not place it in the repository.

Unzip the file “Double.zip” that was downloaded. This has been updated.

Copy the directory to your desktop (Not the repository)

Open this directory and double click the index.html file.

WARNING: Be careful calling other functions. To remove from position 1, if you call pop_front, it will call your remove function and you could end up with each function calling each other. If you wrote the code for the insert and the remove and didn't call a pop or a push, you can quickly write the push and pop functions.