

Linked Lists
Coding Exercise
Due Date is 29-3-2012

Objective: This optional coding exercise is intended to get the student closer to the linked lists implementation and operations using the C++ standard library <list>.

Programming Tools: To code this assignment, compile it, and get it running, you will need to:

1. Download **Microsoft Visual C++ 2008 Express Edition**. Here is the [download link](#).
2. Install the downloaded EXE file of **Microsoft Visual C++ 2008 Express Edition**. This [good link](#) shows how to install **Microsoft Visual C++ 2008 Express Edition** and how to compile and run your C++ programs on it. *Important: during the download and the installation of Microsoft Visual C++ 2008 Express Edition, you will be asked whether you want to install Microsoft SQL server and Microsoft MSDN Library. We don't need those, so please do NOT install them.*

Assignment Requirements:

Given the following two sets of integers:

set1 = { 9, -6, 7, -8, 5, -2, 18, -13, 14}
set2 = {10, -11, 4, -3, 12, -1, -17, 16, -15}

1. Assign the contents of each of the sets to a list (You will need two lists: list1 and list2).
2. Display the contents of list1 one and list2 in a forward order and a reverse order.
3. Create a copy of each list (list3 as a copy of list1, and list4 as a copy of list2). Display list3 and list4 in a reverse order.
4. For list3, replace each negative element with its positive value. Display list3.
5. For list4, replace each positive element with its negative value. Display list4.
6. Sort list1. Display the sorted list.
7. Sort list2. Display the sorted list.
8. Merge list1 and list2 in list1. Display list1 after the merge.

You may start your program for this assignment using the following:

```
#include <iostream>
#include <list>

using namespace std;

int main()
{

    int set1[] = {-9, 6, -7, 8, -5, 2, -18, 13, -14};
    list<int> list1 (set1,set1+9);
    list<int>::iterator it;

    printf("list1 contents:\n");

    for ( it=list1.begin() ; it != list1.end(); it++ )
        printf(" %d",*it);

    printf("\n");

}
```

Once you finish coding, compiling and running your program, the output must match the following:

```
list1 contents:
-9 6 -7 8 -5 2 -18 13 -14
list2 contents:
-10 11 -4 3 -12 1 -17 16 -15
list1 contents (reversed):
-14 13 -18 2 -5 8 -7 6 -9
list2 contents (reversed):
-15 16 -17 1 -12 3 -4 11 -10
list3 contents:
-9 6 -7 8 -5 2 -18 13 -14
list4 contents:
-10 11 -4 3 -12 1 -17 16 -15
list3 contents (no negatives):
9 6 7 8 5 2 18 13 14
list4 contents (no positives):
-10 -11 -4 -3 -12 -1 -17 -16 -15
list1 sorted:
-18 -14 -9 -7 -5 2 6 8 13
list2 sorted:
-17 -15 -12 -10 -4 1 3 11 16
list1 merged with list2:
-18 -17 -15 -14 -12 -10 -9 -7 -5 -4 1 2 3 6 8 11 13 16
```