Chapter VI

Exercise 02: PmergeMe

	Exercise : 02
	PmergeMe

Turn-in directory : *ex02/*

Files to turn in : `Makefile, main.cpp, PmergeMe.{cpp, hpp}`

Forbidden functions : `None`

You must create a program with these constraints:

- The name of the program is PmergeMe.

- Your program must be able to use a positive integer sequence as argument.

- Your program must use the merge-insert sort algorithm to sort the positive integer sequence.

> ⚠ To clarify, yes, you need to use the Ford-Johnson algorithm.

- If an error occurs during program execution, an error message should be displayed on the standard error.

> ⚠ You must use at least two different containers in your code to validate this exercise. Your program must be able to handle at least 3000 different integers.

It is strongly advised to implement your algorithm for each container and thus to avoid using a generic function.

Here are some additional guidelines on the information you should display line by line on the standard output:

- On the first line you must display an explicit text followed by the unsorted positive integer sequence.

- On the second line you must display an explicit text followed by the sorted positive integer sequence.

- On the third line you must display an explicit text indicating the time used by your algorithm by specifying the first container used to sort the positive integer sequence.

- On the last line you must display an explicit text indicating the time used by your algorithm by specifying the second container used to sort the positive integer sequence.

The format for the display of the time used to carry out your sorting is free but the precision chosen must allow to clearly see the difference between the two containers used.

Here is an **example** of a standard use:

```
$> ./PmergeMe 3 5 9 7 4
Before: 3 5 9 7 4
After:  3 4 5 7 9
Time to process a range of   5 elements with std::[..] :  0.00031 us
Time to process a range of   5 elements with std::[..] :  0.00014 us
$> ./PmergeMe `shuf -i 1-100000 -n 3000 | tr "\n" " "`
Before: 141 79 526 321 [...]
After:   79 141 321 526 [...]
Time to process a range of 3000 elements with std::[..] : 62.14389 us
Time to process a range of 3000 elements with std::[..] : 69.27212 us
$> ./PmergeMe "-1" "2"
Error
$> # For OSX USER:
$> ./PmergeMe `jot -r 3000 1 100000 | tr '\n' ' '`
[...]
$>
```

The indication of the time is deliberately strange in this example. Of course you have to indicate the time used to perform all your operations, both the sorting part and the data management part.

Warning: The container(s) you used in the previous exercises are
forbidden here.

The management of errors related to duplicates is left to your
discretion.