

# Algorithms and Iterators Workshop

- This workshop will test your understanding of algorithms and iterators
- As well as giving you practice in using algorithms and iterators, it will give you an appreciation of the power and simplicity of code that can be achieved
- It should be completed using *standard STL algorithms only*, plus lambda functions where appropriate
- *Do not use explicit loops* (for, for\_each, while, do/while) *or recursion*

- You may need to do a little research for some of the questions
- To look up suitable algorithms, or details of how to call them, I recommend the C++ Reference Site:

<https://en.cppreference.com/w/cpp/algorithm>

- 1) Fill a vector with 10 random integers between 0 and 1000
- 2) Display the vector elements on the screen
- 3) Find the maximum element in the vector
- 4) Find the index of this element
  - (hint: subtracting two iterators gives the same result as subtracting the indexes of their elements)
- 5) Sum the elements of the vector

- 6) Normalise the vector (divide all the elements by the largest) and put the results into a vector of doubles, without setting the size of the output vector first
- 7) Count the number of odd numbers in the original vector
- 8) Make a sorted copy of the vector. Without using a functor or a lambda (or equivalent), find the first element greater than 455 and the number of elements  $> 455$
- 9) Copy all the odd numbers to a vector of doubles, without setting the size of the output vector first

- 10) Sort the vector in descending order
- 11) Randomly shuffle all but the first and the last element of the vector
- 12) Remove all the odd numbers from the vector
- 13) Write the remaining elements to a text file on a single line as a comma separated list, without a trailing comma

- 14) Read a file of words and display each word once
  - Ignore space characters, punctuation and capitalization
- 15) Count the total number of words in the file
- 16) Count the number of lines in the file
- 17) Count the number of characters in the file
- 18) Read two files of words and display the words which are common to both files

- 19) Calculate the factorial of 6 ( $6 \times 5 \times 4 \dots \times 1$ )
- 20) Read a word from input and display all the possible permutations of it
- 21) Implement the shuffle algorithm



22) A palindrome is an expression which reads the same backwards, e.g.

- Madam I'm Adam
  - Able was I ere I saw Elba
  - A man, a plan, a canal - Panama!
- Space characters, punctuation and capitalization are ignored
  - Continued on next slide...

- The program prompts the user to enter a palindrome
- Confirm that the input is a palindrome
- If the expression is not a palindrome, print out the first mismatched character and the expression up to that point
- **Sample output:**

```
Please enter your palindrome: Madam I'm Adam  
"Madam I'm Adam" is a palindrome
```

```
Please enter your palindrome: Madam I've Adam  
"Madam I've Adam" is not a palindrome  
First mismatched character is 'm' after "mada"
```