ECS769P: Advanced OOP
Lab 7: STL Iterators and Adaptors

Where appropriate use the Standard Library containers and iterators to solve these problems.

Exercise 1: sizeof (basic)

Write a program that to print out the values of `sizeof(short)`, `sizeof(int)`, `sizeof(long)` and `sizeof(long long)`. Check your answers against other students using different compilers and operating systems.

Exercise 2: Count Occurances (basic)

Write a function that takes a pair of iterators to a `vector<int>` n and an `int`. Look for the value n in that range and return a count of the number of separate occurances. Test your code.

Exercise 3: Capitalisation (core)

Write a function that iterates through a string, capitalising the first letter of each word. It should stop processing and upon reaching an exclamation mark (!). Test your code.

Exercise 4: Binary Search (core)

Write your own binary search function for **built-in integer arrays**, i.e. not STL arrays. It should return true if and only if a given integer is in the array. It should assume the elements are sorted.

Outline: use two pointers to represent the range of elements still to search, and a third pointer as the mid-point of that range. Each iteration checks whether the sought after integer is in the top or bottom half of the range (remember: the elements are sorted!) and redefines the range accordingly.

Write another version of your binary search function that works with `vector<int>`.

Exercise 5: Balanced Brackets (core)

Write a function that checks if a string contains correct brackets, i.e. `()`, {} and `[]` are balanced and nested properly. Use a string iterator and a stack adaptor.

Outline: push opening characters onto the stack as they are processed, and pop when a closing character is encountered.