Practical 1

(Due in Week 2, 2 marks)

Note: *This practical will be assessed in person by your tutor during the scheduled practical session in Week 2. You must complete all tasks prior to your practical class; they should not be completed during the class itself. It is strongly recommended that you bring your own laptop to the session to demonstrate your work directly to your tutor.*

Task 1.1

Consider the definition of the following function template:

```
template <class Type>
void funcExp(Type list[], int size) {
    Type x = list[0];
    Type y = list[size - 1];
    for (int j = 1; j < size; j++) {
        if (x < list[j])
            x = list[j];

        if (y > list[size - 1 -j])
            y = list[size - 1 -j];
    }
    cout << x << endl;
    cout << y << endl;
}
```

Create a `main` function with a declaration of the follow data items:

```
int list[10] = {5,3,2,10,4,19,45,13,61,11};
string strList[] = {"One", "Hello", "Four", "Three", "How", "Six"};
```

and make the following function calls in the main function:

```
funcExp(list, 6);
funcExp(strList, 6);
```

Explain to your tutor the following:

1. the meaning of `template`
2. the meaning of `Type`
3. the running result of your program
4. the outcome of function call `funcExp(list, 10)` without running the program.

Task 1.2

Download the code `Task1_2.zip` from vUWS located in the folder of **Practical** 1. Run the program and provide an explanation of the code to your tutor on the following questions:

- What is the data type of `grid`?
- What does the function `~Board()` do?
- What is the purpose of the function `Board(const Board &cboard)`?

Task 1.3

Change the code of **Task**1.2 so that each cell of the board is filled randomly with either a '`B`', a '`C`' or a ' ' using the function `addMove`. Here is an example of output:

```
    1   2   3   4   5   6   7   8   9
  ---------------------------------
1 |   | C | C | B | B | C |   | C | C |
2 | B |   | B | C |   | B | B | B | B |
3 | C | C | C | C | B | C | B | B | B |
4 | B | B | C | C | C | B |   | C | C |
5 | C | C | B | B | C | B |   | B | B |
6 |   |   | B |   |   | B | B | C | C |
7 | C | B |   |   |   | C | B |   | B |
8 | C | C | B | C |   |   | C | B | B |
9 | B | C | C | B | B | C | C |   | B |
  ---------------------------------
```

You may change the size of board to show different patterns of outputs.

Note: You may make changes on `Board.h`, `Main.cpp` or bother files.

Task 1.4

Change the code of **Task**1.2 so that the program can take inputs from two human players – **Player** `B` and **Player** `C`. If both players choose the same cell, the program takes Player B's move only (in this sense, it is not a turn-taking game but a simultaneous game[1]). Keep taking inputs until the board is full (you may use a 3*3 board to test your program).

[1] This task is a training for assignment 1. Check out Assignment 1 to get more ideas.