Lab 03 - Runtimes

Instructions:

- **The lab requires completing a few tasks.**
- **Your submissions must be submitted to the Lab03 directory of your GitHub repository or uploaded to the Lab03 assignment on Google classroom.**
- **Accompanying these instructions are a few header files that must be included in the appropriate programs you have to write.**
- **Besides the header files provided, your programs can only include the libraries *iostream*, *string*, *fstream*, *sstream*, and *cctype*.**
- **Cheating of any kind is prohibited and will not be tolerated.**
- **Violating and/or failing to follow any of the rules will result in an automatic zero (0) for the lab.**

TO ACKNOWLEDGE THAT YOU HAVE READ AND UNDERSTOOD THE INSTRUCTIONS ABOVE, AT THE BEGINNING OF YOUR SUBMISSION(S), ADD A COMMENT THAT CONSISTS OF YOUR NAME AND THE DATE

Grading:

Task	Maximum Points	Points Earned
1	3	
2	2	
3	5	
Total	10	

Note: solutions will be provided for tasks colored blue only.

Task 1

Create a file named `"main.cpp"` that define an int function named `SumOfTwoMax()` whose header is

```
int SumOfTwoMax(const Array<int>& data)
```

that returns the sum of the two maximum values of *data* if *data* has a size of at least 2; otherwise, it returns 0.

Task 2

Create a file named `"table.xls"` that construct the runtime table of `SumOfTwoMax()` for its worst-case scenario and state what n represents. The table must include a statement column.

Task 3

In the Project.h file from Lab 02, define the string function named `BoardView()` whose header is

```
string BoardView(const string& board,const bool states[])
```

given that *board* is a string of length 36 that consists only of uppercase characters `'A'` through `'R'`, and *states* is a completely initialized bool array of length 36, the function returns a string that is formatted to represents the characters of *board* as a 6 by 6 grid such that

- **the value of every element of *board* are enclosed in square braces in the output string.**
- **if an element of *states* has a value of false, the value of the corresponding element of *board* will be replaced with the character `'X'` in the output string.**

where the elements of *board* and *states* are corresponding if their indices are the same. For instance, a caller of `BoardView()` with the arguments

$$\begin{aligned} \textit{board} &= \texttt{"AQFTHIA}\cdots\texttt{"} \\ \textit{states} &= \{\texttt{false,false,true,false,true,true,false,}\cdots\} \end{aligned}$$

will return

```
"[X][X][F][X][H][I]\n[X]···"
```

where $\cdots$ references to the rest of the characters of the strings and elements of the array.