



**Date handed out: 21/12/2022 7:00 PM Cyprus Time**

**Date submission due: 06/01/2023 23:00 PM Cyprus Time**

## **Indexing Earthquakes**

### **Purpose:**

This assignment aims to help you practice binary search tree ADT, in particular AVL Tree. You will write a program that creates an index of a given list of signatures.

### **Requirements:**

In this assignment, you are given a list of earthquakes in an external file called “earthquakes.txt”. In this text file, you can find a number of real earthquake collected by a digital table.

### **This file includes the following information:**

- **Place:** where the earthquake happened
- **Magnitude:** the size of the earthquake which varies from (2.5 or less up to 8.0 or more) and as the number grows the damage of the earthquake increases.
- **Latitude and Longitude:** represent the exact location of an earthquake.
- **Tsunami:** some earthquakes may lead to a tsunami and accordingly 1 means that there was a Tsunami and 0 means there was not.

And sample data is as follows where each part is separated by “,”

```
Place, Magnitude, Latitude, Longitude, Tsunami
Country2,4.9, -42.2914, 173.8065, 1
Country2,4.6, -17.4442, -174.1893, 0
Country2,5.1, -24.9817, -179.9581, 1
Country2,4.7,6.1803, -82.4554,1
Country7,2.3,38.7666,142.1254,1
Country6,6.3,9.9203,145.3497, 0
Ocean X,3,2.8662,126.6018, 1
```

Your task here is to process this file, and generate an AVL tree based on Place (name of the country or the Ocean). If there is more than one earthquake with the same country or ocean name, they should be then stored together in the same node. Based on this data representation you need to write a program that provides the following functionalities to the user:

1. **Display the full index:** This will display the full AVL tree constructed. For traversal, you need to display the earthquakes sorted based on the place where they happened.

2. **The country with the maximum number of earthquakes:** This will display the details (Place, Magnitude, Latitude, Longitude, and Tsunami) of the earthquakes belonging to the place which has the maximum number of earthquake.
3. **The earthquake with the minimum magnitude:** This will display the details (Place, Magnitude, Latitude, Longitude, and Tsunami) of the earthquake with the minimum magnitude.

### Programming Requirements:

You will start by taking the file name as a command line argument and then you will need to implement the following functions

- **readData():** this function will mainly process the external file. As an input it will take the file name and it will return an AVL tree.
- **insertEarthquake():** This function will take an AVL tree, and the details of an earthquake, and then it will insert the earthquake to the AVL tree. The earthquake will be inserted into the tree based on the place where it happened. You cannot make an assumptions about the number of earthquakes that belongs to the same place. Therefore, if the node with the given place exist then you will add your earthquake details to that node.
- **displayIndex():** This function will mainly take an AVL tree and Display the data in the tree in alphabetical order according to place name of the earthquake.
- **dangerousPlace():** This function will mainly take an AVL tree, will find and display the details of the earthquakes of the place which has the maximum number of earthquakes. In the comment part of this function, discuss the complexity of this function based on your current representation of data. You also need to discuss if there is a way you could improve this.
- **weakestEarthquake():** This function will mainly take an AVL tree and will find and display the earthquake which has the lowest magnitude. In the comment part of this function, discuss the complexity of this function based on your current presentation of data. You also need to discuss if there is a way you could improve this.

Please note, that in this assignment, you can make use of the function in string.h library and similar external libraries. You cannot assume about the number of earthquakes in this external file.

### Submission requirements:

In this assignment, you need to have a header file (avltree.h) which includes the major functionalities of the AVL tree ADT. If you will use other ADTs, you need to create a separate header file for each of them. You also need to have a C source file (indexingEarthquakes.c) that

includes the main function and other functions. You need put all these files into “CNG213a\_3” folder and then submit the compressed version of the folder to ODTUClass. If you do not follow this structure you will lose 10% from the overall grade.

### Programming Style Tips!

Please follow the modular programming approach. In C programming, we use functions referred to modules to perform specific tasks that are determined/guides by the solution.

- Remember the following tips!
- Modules can be written and tested separately!
- Modules can be reused!
- Large project can be developed in parallel by using modules!
- Modules can reduce the length of the program!
- Modules can also make your code more readable!

### Important Notes:

- Remember to have good programming style (Appropriate comments, variable names, formulation of selection statements and loops, reusability, extensibility etc.). Each of the items above will include 10% for good programming style.
- Read rules regarding to assignments from the Syllabus carefully.
- If your code does not compile due to syntax errors, you will automatically get zero.
- If your code includes a variable declaration inside a for loop such as `for(int i=0; i<5;i++)`, you will automatically get zero.
- If your code includes global variables, you will automatically get zero.

### Grading:

Your program will be graded as follows:

Grading Point Mark	(out of 100)
AVL Tree Data Structure	5
Processing data file ( <b>readData()</b> )	15
Inserting/updating a node in the tree ( <b>insertEarthquake()</b> )	25
Display the index ( <b>displayIndex()</b> )	10
Display the details of the earthquakes of the place that has the maximum number of earthquakes ( <b>dangerousPlace()</b> ) (complexity discussion - 3pts)	15
Display the earthquake which has the minimum magnitude ( <b>weakestEarthquake()</b> ) (complexity discussion - 3pts)	15
The main function	15

### Sample Run:

you can find the sample run of the program for the first 4 entries in the above text file.

**Welcome to Earthquake Indexing :) :)**

---

#### Menu

1. Display the full index of earthquakes
  2. Display the earthquakes of the most dangerous place
  3. Display the weakest earthquake
  4. Exit
- 

#### Enter your option: 1

Country2;4.9; -42.2914; 173.8065; 1  
Country2;4.6; -17.4442; -174.1893; 0  
Country2;5.1; -24.9817; -179.9581; 1  
Country2;4.7;6.1803; -82.4554; 1  
Country7;2.3;38.7666;142.1254;1  
Country6;6.3;9.9203;145.3497; 0  
Ocean X;3;2.8662;126.6018; 1

---

#### Menu

1. Display the full index of earthquakes
  2. Display the earthquakes of the most dangerous place
  3. Display the weakest earthquake
  4. Exit
- 

#### Enter your option: 2

Country2;4.9; -42.2914; 173.8065; 1  
Country2;4.6; -17.4442; -174.1893; 0  
Country2;5.1; -24.9817; -179.9581; 1  
Country2;4.7;6.1803; -82.4554; 1

---

#### Menu

1. Display the full index of earthquakes
  2. Display the earthquakes of the most dangerous place
  3. Display the weakest earthquake
  4. Exit
- 

#### Enter your option: 3

##### Weakest Earthquake:

Country7;2.3;38.7666;142.1254;1

---

#### Menu

1. Display the full index of earthquakes

2. Display the earthquakes of the most dangerous place
  3. Display the weakest earthquake
  4. Exit
- 

**Enter your option: 3**

**bye bye !!!**