Sean Poston

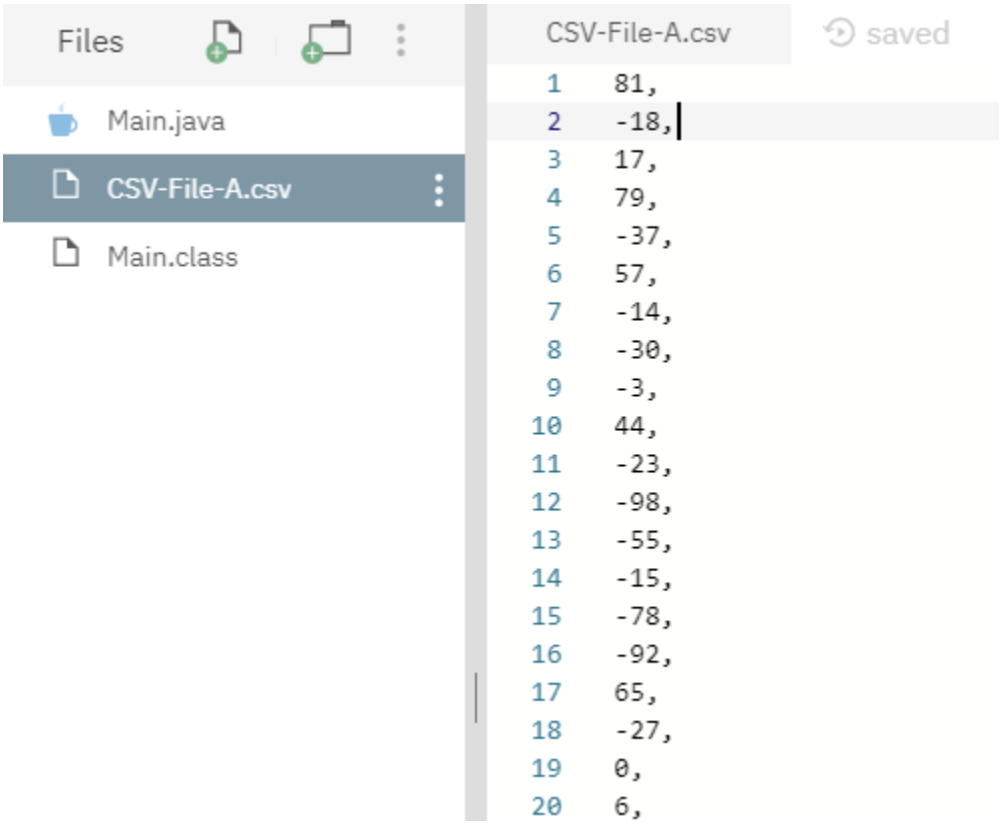
CS300

2/15/2020

```
/*
Author: Sean Poston
Purpose: To write a CS file
Date: 2/14/2020
*/
import java.io.*;
public class Main {
    public static void main(String[] args) {
        try {
            File fileOut = new File("CSV-File-A.csv");
            fileOut.createNewFile(); //create a new file to write to

            FileWriter writer = new FileWriter(fileOut);
            for (int i = 0; i < 50; i++) { //loop to write random numbers to the
.csv file
                int x = (int)((Math.random() * 200) - 100);
                writer.write(x + ",\n");
            }

            writer.close(); //close the files.
        }

        catch(Exception c){
            System.out.println(c); //general catch for different exceptions
        }
    }
}
```

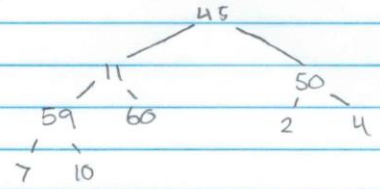
https://repl.it/@seanposton4/Assignment-3

Output:


Files
Main.java
CSV-File-A.csv
Main.class
CSV-File-A.csv
saved
1 81,
2 -18,
3 17,
4 79,
5 -37,
6 57,
7 -14,
8 -30,
9 -3,
10 44,
11 -23,
12 -98,
13 -55,
14 -15,
15 -78,
16 -92,
17 65,
18 -27,
19 0,
20 6,

CS-300
Sean Poston

23.6.6 The heap will be created in the order that it's placed in the list. Each index can be found by the index = i. This means that children of that index will be (2i+1) and (2i+2) for the left and right child respectively. This means the heap will look like this:

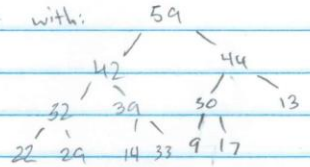


23.6.7 Assuming you want to remove the root:

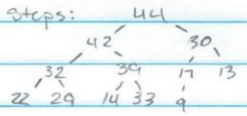
1) you remove the largest number (62) and replace it with the last node (9).

2) You then swap 9 with the largest number until it's the smallest number

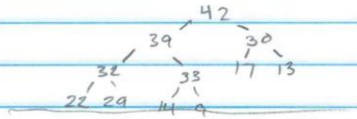
We are left with:



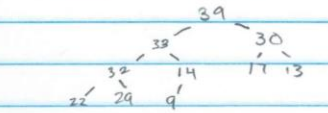
repeating these steps:



again:



again:



again: