HW1_P1 - Largest of three numbers

Student name :	Carter Hawks
Student email :	ckh170000@utdallas.edu
Class name :	2336.001_F18
Submitted on :	Sep 05, 2018 04:34 pm

solution.cpp

```
/* Write your Analysis here

Given three numbers, logically determine the greatest of the set.

*/
/* Write your Design here

1. Input numbers into an array in order to make them easier to work with
2. Sort array
    (ascending vs descending doesn't really do anything,
        just changes how logic is written)
3. Begin to numerical comparisons
    - Check if all numbers are equal
    - Check if two numbers are equal, and then if the third number is
        greater or not
    - If there are no equals, return the highest (determined easily from
        the sorted array)


*/
// Write your code here
#include<iostream>
#include<cstdio>
#include <bits/stdc++.h>

using namespace std;

int main(){
    // define nums
    int N1;
    int N2;
    int N3;
    // take in user input nums
    cin >> N1;
    cin >> N2;
    cin >> N3;
    // put input nums into array and sort ascending
    int nums[] = { N1, N2, N3 };
    sort(nums, nums + 3);
```

```
    // debug for making sure the sort was actually working
    for(int i = 0; i < 3; i++){
        //cout << nums[i] << " ";
    }

    // if all numbers are equal
    if(nums[0] == nums[1] && nums[1] == nums[2]){
        cout << 1 << endl;
    }
    // if there are no equal numbers, the last one in the array has to be the greatest
    else if(nums[0] < nums[1] && nums[1] < nums[2]){
        cout << nums[2] << endl;
    }
    // if the greater two numbers are equal, and first number is lower
    else if(nums[0] < nums[1] && nums[1] == nums[2]){
        cout << 0 << endl;
    }
    // if the least two numbers are equal, and last number is greater
    else if(nums[0] == nums[1] && nums[1] < nums[2]){
        cout << nums[2] << endl;
    }

    return 0;
}
```

Name

Custom test case

Input

3 3 9

Output (Lines:2)

9

Expected Output (Lines:0)

Status

NA

Name

Custom test case

Input

654 765 3

Output (Lines:2)

765

Expected Output (Lines:0)

Status
NA

Name
Custom test case

Input
1 2 3

Output (Lines:2)
3

Expected Output (Lines:0)

Status
NA

Name
Default

Input
5 6 7

Output (Lines:2)
7

Expected Output (Lines:1)
7

Status
Pass
