

Smallest Two

My algorithm has two values, second and smallest each initially set to max int. Every input is taken and compared to the smallest and second smallest. If it is smaller than the smallest thus far it replaces the smallest, and the previous smallest replaces the second. If it is smaller than the second it checks to ensure it isn't equal to the smallest before replacing the second.

Pseudocode:

```
int smallest = INFINITY
int second = INFINITY

for each input in stdin:
    if( input < smallest ){
        second = smallest
        smallest = input
    }elif (input < second && input != smallest ){
        second = input
    }
```

This is correct because if the input most recently read is larger than the two recorded values it is ignored. And if it is found to be smaller than either they are rearranged accordingly. Every input is compared to the smallest and second smallest therefore none can be missed.

This runs in $O(n)$ time because it takes only one iteration through every input to determine the two smallest.