

Our Solution(s)Run Code

Solution 1

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 #include <vector>
4 using namespace std;
5
6 class MinHeap {
7 public:
8     vector<int> heap;
9
10    MinHeap(vector<int> vector) { heap = buildHeap(&vector); }
11
12    // O(n) time | O(1) space
13    vector<int> buildHeap(vector<int> *vector) {
14        int firstParentIdx = (vector->size() - 2) / 2;
15        for (int currentIdx = firstParentIdx; currentIdx >= 0; currentIdx--) {
16            siftDown(currentIdx, vector->size() - 1, vector);
17        }
18        return *vector;
19    }
20
21    // O(log(n)) time | O(1) space
22    void siftDown(int currentIdx, int endIdx, vector<int> *heap) {
23        int childOneIdx = currentIdx * 2 + 1;
24        while (childOneIdx <= endIdx) {
25            int childTwoIdx = currentIdx * 2 + 2 <= endIdx ? currentIdx * 2 + 2 : -1;
26            int idxToSwap;
27            if (childTwoIdx != -1 && heap->at(childTwoIdx) < heap->at(childOneIdx)) {
28                idxToSwap = childTwoIdx;
29            } else {
30                idxToSwap = childOneIdx;
31            }
32            if (heap->at(idxToSwap) < heap->at(currentIdx)) {
33                swap(heap->at(currentIdx), heap->at(idxToSwap));
34                currentIdx = idxToSwap;
35                childOneIdx = currentIdx * 2 + 1;
36            }
37        }
38    }
```

Your SolutionsRun Code

Solution 1Solution 2Solution 3

```
1 #include <vector>
2 using namespace std;
3
4 // Do not edit the class below except for the buildHeap,
5 // siftDown, siftUp, peek, remove, and insert methods.
6 // Feel free to add new properties and methods to the class.
7 class MinHeap {
8 public:
9     vector<int> heap;
10
11    MinHeap(vector<int> vector) { heap = buildHeap(&vector); }
12
13    vector<int> buildHeap(vector<int> *vector) {
14        // Write your code here.
15        return *vector;
16    }
17
18    void siftDown(int currentIdx, int endIdx, vector<int> *heap) {
19        // Write your code here.
20    }
21
22    void siftUp(int currentIdx, vector<int> *heap) {
23        // Write your code here.
24    }
25
26    int peek() {
27        // Write your code here.
28        return -1;
29    }
30
31    int remove() {
32        // Write your code here.
33        return -1;
34    }
35}
```

```

1 class Program {
2     public static void Main()
3     {
4         Console.WriteLine("Enter two numbers:");
5         int a = int.Parse(Console.ReadLine());
6         int b = int.Parse(Console.ReadLine());
7         int sum = a + b;
8         int difference = a - b;
9         int product = a * b;
10        int quotient = a / b;
11        int remainder = a % b;
12        Console.WriteLine("Sum: {0}, Difference: {0}, Product: {0}, Quotient: {0}, Remainder: {0}", sum, difference, product, quotient, remainder);
13    }
14 }

```

Run or submit code when you're ready.