
Đã bắt đầu vào lúc Thứ bảy, 15 Tháng mười 2022, 5:58 PM

Tình trạng Đã hoàn thành

Hoàn thành vào lúc Thứ bảy, 29 Tháng mười 2022, 8:04 PM

Thời gian thực hiện 14 ngày 2 giờ

Điểm 15,00 của 15,00 (100%)

Câu hỏi 1

Chính xác

Điểm 1,00 của 1,00

Implement all methods in class **Queue** with template type **T**. The description of each method is written as comment in frame code.

```
#ifndef QUEUE_H
#define QUEUE_H
#include "DLinkedList.h"
template<class T>
class Queue {
protected:
    DLinkedList<T> list;
public:
    Queue() {}
    void push(T item) ;
    T pop() ;
    T top() ;
    bool empty() ;
    int size() ;
    void clear() ;
};
#endif /* QUEUE_H */
```

You can use all methods in class **DLinkedList** without implementing them again. The description of class **DLinkedList** is written as comment in frame code.

```
template <class T>
class DLinkedList
{
public:
    class Node;      //forward declaration
protected:
    Node* head;
    Node* tail;
    int count;
public:
    DLinkedList() ;
    ~DLinkedList();
    void add(const T& e);
    void add(int index, const T& e);
    T removeAt(int index);
    bool removeItem(const T& removeItem);
    bool empty();
    int size();
    void clear();
    T get(int index);
    void set(int index, const T& e);
    int indexOf(const T& item);
    bool contains(const T& item);
};
```

For example:

Test	Result
Queue<int> queue; assert(queue.empty()); assert(queue.size() == 0);	

Answer: (penalty regime: 0, 0, 5, 10 %)

Reset answer

```
1 void push(T item) {
2     // TODO: Push new element into the end of the queue
3     list.add(item);
4 }
5
6 T pop() {
7     // TODO: Remove an element in the head of the queue
8     T ret=list.get(0);
9     list.removeAt(0);
10    return ret;
11 }
12
13 T top() {
14     // TODO: Get value of the element in the head of the queue
15     T ret=list.get(0);
16     return ret;
17 }
18
19 bool empty() {
20     // TODO: Determine if the queue is empty
21     return list.isEmpty();
22 }
23
24
25 int size() {
26     // TODO: Get the size of the queue
27     return list.size();
28 }
29
30 void clear() {
31     // TODO: Clear all elements of the queue
32     list.clear();
33 }
```

Passed all tests! ✓

(Chính xác)

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 2

Chính xác

Điểm 1,00 của 1,00

A nice number is a positive integer that contains only 2's and 5's.

Some nice numbers are: 2, 5, 22, 25, 52, 55, ...

Number 2 is the first nice number.

Given an integer N, return the Nth nice number.

Note: iostream, vector, queue are already included for you.

Constraint:

$1 \leq n \leq 10^6$

Example 1:

Input:

n = 5

Output:

52

Explanation:

The sequence of nice numbers is 2, 5, 22, 25, 52, 55, ...

The 5th number in this sequence is 52

Example 2:

Input:

n = 10000

Output:

2255522252225

For example:

Test	Input	Result
int n; cin >> n; cout << nthNiceNumber(n) << endl;	5	52
int n; cin >> n; cout << nthNiceNumber(n) << endl;	10000	2255522252225

Answer: (penalty regime: 0, 0, 0, 5, 10, ... %)

Reset answer

```

1 // iostream, vector and queue are included
2 // You can write helper methods
3
4 long long nthNiceNumber(int n) {
5     queue<long long> q;
6     q.push(2);
7     q.push(5);
8     for(int i=1;i<n;i++){
9         long long temp=q.front();
10        q.pop();
11        q.push(temp*10+2);
12        q.push(temp*10+5);
13    }
14    return q.front();
15 }
```



	Test	Input	Expected	Got	
✓	int n; cin >> n; cout << nthNiceNumber(n) << endl;	5	52	52	✓
✓	int n; cin >> n; cout << nthNiceNumber(n) << endl;	10000	2255522252225	2255522252225	✓

Passed all tests! ✓

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 3

Chính xác

Điểm 1,00 của 1,00

A group of N students in HCMUT are playing a funny game. They gather around a circle and number themselves from 1 to N clockwise. After a step of the game, a person is removed from the circle. The last person to stay in the circle is the winner.

The game's rule is as follows:

1. The game start at the person numbered 1.
2. From the current person, count k people clockwise (including the person you started at). The counting may wraps around the circle.
3. The last counted one is remove from the circle.
4. If the circle still has more than one people, the game continues from the person immediately clockwise of the person who just lost the game.
Then repeat step 2
5. The last person in the game will win.

Toan really wants to win the game to impress their friends. Given the number of players, N, and an integer, k. Help Toan win the game by determine the number in which he has to be standing to certainly win the game.

Hint: You can use a queue to simulate the process of the game.

Constraint:

$1 \leq k \leq N \leq 10^4$

Example:

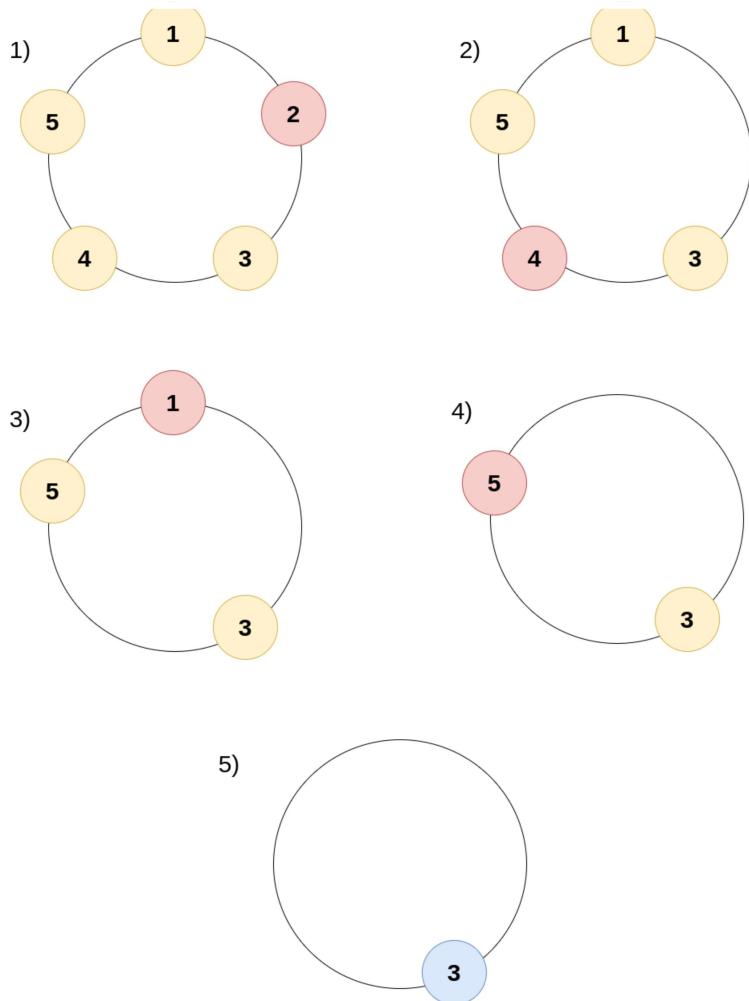
Input:

$n = 5, k = 2$

Output

3

Explanation: The game proceeds as follows



For example:

Test	Input	Result
<pre>int N; int k; cin >> N >> k; cout << numberOfTheWinner(N, k);</pre>	5 2	3
<pre>int N; int k; cin >> N >> k; cout << numberOfTheWinner(N, k);</pre>	6 5	1

Answer: (penalty regime: 0, 0, 0, 5, 10, ... %)

Reset answer

```

1 // iostream and queue are included
2 // Hint: Use a queue to simulate the process
3
4 int numberOfTheWinner(int N, int k) {
5     queue<int> q;
6     for(int i=1;i<=N;i++){
7         q.push(i);
8     }
9     while(q.front()!=q.back()){
10        int x=k-1;
11        while(x>0){
12            int temp=q.front();
13            q.pop();
14            q.push(temp);
15            x--;
}

```

```
16 }  
17     q.pop();  
18 }  
19 return q.front();  
20 }
```

	Test	Input	Expected	Got	
✓	int N; int k; cin >> N >> k; cout << numberOfTheWinner(N, k);	5 2	3	3	✓
✓	int N; int k; cin >> N >> k; cout << numberOfTheWinner(N, k);	6 5	1	1	✓

Passed all tests! ✓

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 4

Chính xác

Điểm 1,00 của 1,00

Given an array of integers.

Your task is to implement a function with following prototype:

```
int sumOfMaxSubarray(vector<int>& nums, int k);
```

The function returns the sum of the maximum value of every subarray of `nums` with fixed length `k`.

Note:

- The `iostream`, `vector`, `queue` and `deque` libraries have been included and `namespace std` is being used. No other libraries are allowed.
- You can write helper functions and classes.

For example:

Test	Result
<pre>vector<int> nums {1, 2, 4, 3, 6}; int k = 3; cout << sumOfMaxSubarray(nums, k);</pre>	14

Answer: (penalty regime: 0 %)

Reset answer

```

1 int sumOfMaxSubarray(vector<int>& nums, int k) {
2     deque<int> d; // deque to store INDICES of Maximums (usually in the front of deque)
3
4     // deque elements would be arranged:
5     // Max --- Min
6
7     int sum = 0;
8
9     // Push in maximum of the first subarray
10    int i = 0;
11    while (i < k)
12    {
13        while (!d.empty() && nums.at(i) >= nums.at(d.back()))
14            d.pop_back();
15
16        d.push_back(i);
17        ++i;
18    }
19    sum += nums.at(d.front()); // Front is always a maximum value of the current subarray
20
21    // Continue with other elements
22    // nums[i] is new element added
23    while (i < (int)nums.size())
24    {
25        // Pop old elements
26        if (!d.empty() && d.front() < i - k + 1)
27            d.pop_front();
28
29        while (!d.empty() && nums.at(i) >= nums.at(d.back()))
30            d.pop_back();
31
32        d.push_back(i);
33        ++i;
34
35        sum += nums.at(d.front());
36    }
37
38    return sum;
39 }
```

	Test	Expected	Got	
✓	vector<int> nums {1, 2, 4, 3, 6}; int k = 3; cout << sumOfMaxSubarray(nums, k);	14	14	✓
✓	vector<int> nums {8016}; int k = 1; cout << sumOfMaxSubarray(nums, k);	8016	8016	✓
✓	vector<int> nums {217, 3176, 5193, 1292, 2720, 6042, 7769, 6965, 5782, 9595, 4259, 2564, 1475, 2397, 2252, 789, 6762, 1116, 5927, 3859}; int k = 9; cout << sumOfMaxSubarray(nums, k);	107648	107648	✓
✓	vector<int> nums {2539, 7651, 429, 6382, 17, 5187, 1517, 1052, 8964, 8542, 7212, 2895, 5444, 7976, 6976, 9629, 1404, 6577, 5784, 5085, 5652, 7853, 138, 3372, 8458, 1747, 7722, 9725, 3490, 2298, 7694, 4800, 1403, 961, 6255, 1431, 2372, 3117, 3619, 8453, 2320, 5800, 1913, 3848, 3350, 4776, 6756, 6889, 2251, 3758, 7132, 6002, 1412, 3616, 5665, 783, 3063, 4186, 9025, 599, 6913, 4384, 2700, 7581, 2093, 7719, 4326, 274, 7299, 5941, 7054, 2557, 5600, 421, 8204, 3454, 5029, 447, 1488, 127, 5442, 5864, 1240, 492, 3652, 5574, 802, 5861, 6876, 2999, 8131, 1121, 3298, 8035, 2295, 1525, 1738, 3662, 2859, 8150}; int k = 74; cout << sumOfMaxSubarray(nums, k);	262575	262575	✓
✓	vector<int> nums {3986, 2422, 1451, 1660, 5869, 1119, 342, 7241, 9063, 7147, 4069, 3910, 5047, 9846, 6004, 9914, 1430, 2324, 933, 6641, 1684, 1672, 4051, 2556, 5545, 2245, 9219, 3544, 4591, 7108, 3019, 7667, 2334, 2677, 1704, 5596, 3976, 1656, 4171, 1979, 8375, 4830, 6387, 3905, 6291, 3420, 2081, 8065, 872, 7665, 2968, 6879, 4871, 7784, 5940, 455, 1196, 1744, 7884, 6801, 7441, 5529, 2171, 5444, 5280, 1425, 3451, 1353, 3870, 9553, 2851, 5394, 3655, 754, 9395, 9450, 180, 6707, 8220, 9502, 36, 3839, 6342, 5421, 4490, 571, 4364, 3532, 179, 1262, 2155, 5804, 3675, 852, 1147, 464, 801, 4628, 2541, 3418, 279, 1584, 854, 7798, 3854, 9020, 4506, 7881, 7362, 7651, 3245, 8715, 1506, 215, 70, 3411, 5300, 8408, 2389, 1409, 5206, 231, 8534, 8162, 3127, 4539, 6518, 6751, 1623, 596, 8389, 6824, 6300, 7726, 909, 9199, 4374, 7783, 6082, 2615, 4086, 5700, 4235, 8191, 9851, 9585, 7808, 2537, 428, 1177, 2404, 1713, 9901, 4901, 3701, 330, 1228, 4023, 3106, 9722, 1387, 2762, 1660, 2681, 6787, 9458, 4000, 9302, 2598, 1386, 3123, 4843, 9430, 1806, 6527, 9042, 6085, 8789, 622, 5032, 3068, 9188, 2292, 3457, 9872, 3871, 8977, 1004, 705, 1008, 9527, 8952, 1253, 5213, 6586, 7111, 3269, 4230, 7670, 9202, 2358, 988, 9269, 9070, 8075, 5656, 9248, 7613, 3209, 8091, 4213, 1975, 9243, 2561, 955, 8123, 7175, 9164, 3090, 497, 9888, 3166, 2654, 9250, 7758, 8855, 342, 7850, 1803, 9296, 2183, 2460, 3934, 1174, 5567, 3110, 1998, 8444, 9783, 1639, 7903, 7338, 4464, 9455, 7809, 4618, 8321, 5814, 8618, 6853, 6639, 6896, 2704, 2441, 5919, 6814, 492, 2478, 1976, 7640, 6811, 9577, 9901, 9563, 2359, 576, 7705, 8093, 1541, 4560, 2075, 1321, 6064, 4267, 9124, 9639, 8758, 2950, 8111, 1866, 1921, 3666, 7633, 8833, 8681, 3976, 8241, 310, 9960, 9995, 713, 2539, 15, 9801, 5901, 2391, 6343, 8482, 6012, 2923, 2650, 8204, 5557, 2713, 5570, 1734, 588, 5239, 3842, 1116, 9533, 3008, 4021, 4219, 4255, 8252, 1875, 2725, 2333, 9794, 347, 438, 7649, 1915, 4269, 5128, 6290, 8056, 3815, 2068, 8224, 2582, 1331, 6778, 685, 5409, 751, 9411, 4190, 2279, 5949, 1413, 1122, 1199, 3836, 2711, 6562, 410, 247, 8534, 6394, 2248, 8892, 7959, 1287, 1214, 2495, 7757, 962, 2295, 9152, 2557, 2252, 9832, 3658, 3649, 4474, 2088, 9614, 394, 1603, 96, 799, 3177, 4134, 4256, 9503, 8463, 2018, 3874, 9051, 6596, 721, 2658, 2674, 9061, 4772, 894, 444, 5857, 837, 6075, 1480, 7367, 2345, 9049, 6838, 1992, 6660, 3431, 7686, 2898, 210, 5801, 4177, 6176, 3619, 7993, 71, 5916, 9958, 4989, 3608, 2534, 5561, 6328, 4599, 5370, 1424, 7490, 3931, 9101, 5092, 4524, 840, 3496, 1230, 4079, 2607, 8184, 6308, 5973, 4177, 1956, 3731, 2587, 233, 7217, 8415, 9620, 2186, 4628, 2662, 8945, 3142, 710, 4955, 9328, 2946, 3628, 6434, 851, 9910, 3127, 9463, 6931, 9257, 296, 2436, 388, 6604, 105, 5202, 4250, 1974, 6025, 8832, 1420, 416, 6310, 2850, 505, 8626, 2047, 4284, 8712, 4174, 2522, 9503, 1892, 3514, 5700, 2402, 8600, 5396, 615, 2405, 5047, 738, 7777, 738, 3983, 1767, 1603, 1686, 4191, 4253, 374, 6814, 4886}; int k = 275; cout << sumOfMaxSubarray(nums, k);	2257701	2257701	✓
✓	vector<int> nums {501, 581, 14, 3557, 2439, 3613, 5034, 3286, 9803, 5292, 2875, 3017, 8570, 9337, 4441, 7469, 215, 1703, 2016, 7434, 5530, 4481, 1417, 1095, 797, 2358, 1419, 9566, 934, 856, 577, 2492, 7271, 2447, 3373, 8349, 3490, 6852, 3037, 5432, 4193, 6666, 1902, 2918, 4341, 9119, 2623, 8809, 3773, 7860, 6215, 1787, 8266, 4953, 9784, 5956, 1776, 381, 2950, 3985, 5984, 7076, 3859, 2513, 3908, 6922, 1037, 8129, 9706, 374, 7323, 8394, 532, 7299, 7220, 1129, 7471, 9028194}	9028194	9028194	✓

Test	Expected	Got	
1035, 3472, 464, 179, 7941, 1875, 9916, 8586, 8081, 9495, 3319, 5669, 1633, 3800, 7262, 8923, 698, 1718, 149, 899, 9659, 2087, 7241, 9013, 9662, 1340, 2790, 2396, 6348, 6773, 9242, 7534, 1782, 2326, 8010, 2591, 33, 1228, 1621, 1442, 7202, 7035, 7446, 6802, 5786, 4842, 1133, 5504, 9891, 1682, 6543, 238, 5644, 3602, 2013, 3387, 7602, 1846, 4835, 4864, 4000, 2665, 6370, 1995, 8231, 601, 8160, 9065, 481, 7898, 1454, 6906, 4755, 3902, 970, 5471, 6306, 9411, 1014, 9367, 9159, 6285, 1209, 5419, 5349, 1286, 9685, 998, 7258, 6147, 8040, 7289, 1461, 6697, 9767, 670, 823, 256, 7688, 235, 6653, 5206, 3183, 169, 8893, 3207, 6587, 1782, 4910, 540, 9841, 9272, 5949, 6913, 954, 3139, 5806, 540, 2249, 4061, 3567, 9322, 4287, 9455, 963, 7596, 4210, 2958, 8478, 2099, 6203, 2700, 2476, 631, 1488, 8777, 4103, 302, 2594, 1805, 4345, 5419, 5927, 3562, 5118, 8261, 5418, 7134, 9533, 3690, 913, 5447, 8873, 2532, 8287, 954, 1299, 3786, 923, 4453, 1987, 1946, 6146, 1795, 1831, 1350, 5801, 8160, 586, 2156, 117, 2046, 1144, 5402, 5102, 1239, 625, 34, 5377, 7108, 2845, 5530, 565, 5917, 2927, 3016, 7584, 1810, 1213, 7852, 2043, 7561, 1246, 8926, 8323, 1593, 693, 4233, 2002, 6103, 4878, 7976, 3117, 4351, 7334, 4305, 2415, 6421, 9375, 8015, 3367, 1500, 308, 6708, 5309, 279, 7354, 3022, 5322, 9086, 1544, 9305, 7224, 9811, 4633, 5478, 840, 7670, 7759, 4779, 7955, 8278, 6458, 5818, 1155, 1001, 1497, 2870, 3827, 45, 144, 4068, 8596, 5791, 5818, 8639, 1233, 9533, 1185, 878, 4709, 3047, 3085, 7353, 2572, 1083, 4089, 4586, 7399, 4338, 8284, 3713, 2436, 2708, 6678, 7051, 3854, 3011, 473, 6003, 48, 3703, 9115, 7833, 2032, 5893, 457, 5827, 8106, 1494, 3629, 5457, 1785, 997, 8723, 2597, 7869, 7449, 41, 5315, 7306, 180, 8549, 8693, 8974, 9771, 7067, 818, 6437, 2903, 2127, 5536, 1008, 4107, 7112, 2263, 5779, 9222, 166, 3544, 446, 1867, 5766, 1710, 959, 3440, 4067, 868, 4885, 3493, 3711, 4504, 2696, 1445, 7554, 8766, 7543, 7729, 8257, 6807, 6962, 7467, 5347, 2995, 314, 320, 9600, 853, 2005, 234, 844, 5687, 3686, 1477, 3315, 2366, 6701, 2384, 1284, 2158, 7371, 2758, 7571, 5073, 5769, 9726, 6394, 5444, 7337, 9967, 6591, 6323, 6316, 6696, 44, 6838, 789, 6527, 4987, 411, 5711, 1049, 4944, 3079, 8011, 730, 2172, 695, 6316, 4108, 5263, 7852, 2759, 69, 430, 486, 7092, 4276, 1653, 4400, 3994, 4989, 1982, 2909, 987, 4854, 7155, 9680, 7243, 4800, 8016, 1058, 7495, 6362, 5949, 9008, 7283, 6226, 107, 1587, 2709, 5326, 2988, 4565, 4928, 1978, 1271, 1984, 4531, 1394, 8172, 1001, 2372, 883, 2822, 8957, 295, 9405, 8639, 6602, 931, 6795, 6376, 3457, 790, 8534, 5593, 923, 120, 5847, 2293, 9402, 8846, 2556, 6930, 5841, 2512, 781, 8184, 2422, 7085, 575, 7351, 4775, 6541, 6392, 6606, 1867, 3624, 7620, 4713, 6843, 9557, 2056, 2395, 1585, 4629, 9733, 9404, 3199, 3281, 895, 4085, 114, 6065, 2751, 5050, 3848, 7822, 7733, 7734, 9665, 1050, 5586, 6077, 4947, 9791, 9845, 5470, 656, 8726, 4437, 2152, 3803, 2809, 2478, 272, 2728, 2570, 5327, 3935, 7140, 3814, 5028, 9936, 245, 5764, 9903, 4635, 8755, 2386, 908, 59, 2120, 5554, 8932, 7799, 7667, 1857, 227, 1727, 9347, 3417, 7366, 5989, 6114, 9591, 7847, 4383, 4720, 4847, 4956, 1226, 3746, 6910, 7020, 1713, 4091, 4945, 3971, 1230, 9353, 6948, 6264, 5471, 9811, 862, 4621, 1453, 6018, 1691, 6104, 9561, 5277, 1113, 1584, 9512, 8265, 9719, 297, 3745, 5936, 3451, 1459, 2644, 8178, 5283, 5231, 1454, 4794, 7902, 6421, 1402, 4131, 4237, 7650, 7226, 2234, 1764, 9861, 2534, 5, 7461, 3932, 3331, 9092, 2526, 2075, 9800, 2743, 7761, 9853, 4782, 8065, 1510, 1949, 2278, 2523, 2556, 1447, 5839, 508, 2837, 7 ...snip... 79, 7297, 7244, 5269, 1111, 4748, 8912, 6363, 8455, 1789, 2653, 5386, 4458, 1618, 8661, 4922, 396, 8470, 1513, 879, 7727, 4037, 9060, 5425, 1917, 8111, 7320, 9821, 5487, 428, 782, 8042, 4413, 2511, 1368, 1838, 3879, 4350, 2106, 1058, 1795, 8544, 5336, 3108, 280, 5064, 449, 9835, 3163, 2017, 6061, 1424, 8360, 667, 7602, 2011, 1939, 533, 2477, 2779, 1471, 9611, 527, 6346, 4591, 5986, 2422, 4624, 8667, 3105, 943, 3976, 725, 1778, 5000, 4480, 7981, 8218, 922, 1428, 9378, 2437, 7271, 7549, 6170, 4006, 4064, 2390, 6851, 93, 346, 752, 7418, 2809, 6514, 9991, 3341, 6753, 5967, 2372, 2972, 4574, 167, 6755, 661, 1739, 4629, 3208, 1873, 3567, 3577, 8478, 1528, 2154, 6511, 8880, 4940, 894, 6117, 7276, 1013, 208, 568, 3012, 2008, 9407, 1550, 4279, 4014, 5717, 3642, 1120, 1330, 8876, 7611, 6413, 2139, 5010, 517, 8594, 2235, 7392, 532, 4511, 8295, 1623, 2926, 3408, 4424, 1964, 8361, 8291, 1137, 7238, 6094, 6127, 5425, 7045, 8140, 8195, 2315, 2262, 7631, 5277, 2549, 4388, 6338, 4716, 2511, 6656, 9506, 8, 6388, 4228, 3605, 9504, 5173, 9781, 1740, 9188, 1412, 3336, 9557, 7349, 6286, 3878, 420, 2266, 9357, 7154, 7410, 2708, 9069, 6571, 662, 6547, 333, 7741, 9108, 5418, 2552, 845, 7283, 9649, 7621, 6868, 4308, 2650, 2779, 7435, 8719, 7110, 8665, 4921, 5392, 4505, 645, 5769, 5890, 1963, 5924, 2191, 5722, 2464, 4682, 7308, 1544, 5002, 3729, 2128, 73, 5950, 2623, 9152, 4545, 853, 724, 4664, 8311, 9587, 2385, 34, 7409, 893, 4911, 424, 4018, 7335, 3518, 422, 7089, 283, 9821, 100, 3596, 9633, 2986, 2242, 6860, 779, 2219, 6229, 3082, 192, 972, 7793, 5299, 1484, 9225, 5223, 984, 5173, 9311, 434, 201, 2625, 3150, 9036, 7748, 1558, 9329, 8294, 7448, 659, 6177, 920, 5710, 1757, 4631, 9317, 6506, 9612, 9553, 5283, 8911, 9544, 8199, 393, 9633, 7774, 2622, 7178, 1855, 6247, 5068, 6130, 32, 8922, 7885, 974, 3052, 416, 4854, 3408, 8175, 4889, 4392, 1875, 1978, 3902, 919, 5580, 6459, 2839, 2684, 2786, 5262, 782, 24, 7984, 5411, 6019, 4681, 184, 2914, 560, 754, 4794, 381, 304, 6822, 5781, 8869, 8060, 7525, 8269, 6559, 6535, 9834, 2286, 935, 347, 2740, 1024, 7702, 9090, 8372, 492, 6769, 970, 54, 3469, 6578, 883, 676, 4922, 4779, 3386, 9815, 3203, 2504, 1819, 1935, 5722, 5524, 6162, 1589, 4485, 3199, 3059, 2143, 9729, 814, 9052, 1636, 281, 2835, 5367, 1040, 1174, 4991, 8084, 8714, 9929, 4853, 4326, 1845, 328, 3297, 908, 820, 1077, 2041, 6883, 3937, 4085, 8579, 7307, 1951, 1071, 3703, 9223, 4158, 7267, 9542, 7193, 5967, 2622, 6699, 1501, 7690, 6939, 1184, 2084, 9852, 2315, 8037, 9444, 5396, 4665, 8477, 1294, 2992, 7239, 6873, 4513, 5044, 717,			

Test	Expected	Got	
<pre> 4985, 3293, 5867, 1684, 2528, 5386, 8528, 4525, 1975, 1161, 4751, 1959, 2981, 7388, 5752, 2609, 765, 2146, 6900, 5756, 1704, 3861, 8347, 6149, 7254, 9160, 5141, 3458, 8191, 8251, 3439, 955, 5854, 7905, 5713, 155, 7440, 7508, 5565, 2150, 7130, 9335, 2123, 9613, 5913, 1291, 4163, 4032, 1613, 1126, 1178, 7252, 3049, 6338, 8144, 2325, 724, 8338, 8425, 605, 499, 5546, 8812, 3926, 299, 8607, 6216, 9421, 8450, 1613, 2621, 6279, 3717, 3768, 1802, 1544, 271, 4308, 3363, 849, 7465, 6633, 3589, 3978, 8733, 2929, 6509, 6225, 7983, 4644, 7429, 2497, 4382, 1143, 9585, 9783, 519, 7616, 2762, 8129, 9947, 9324, 8552, 9608, 7378, 2965, 8165, 1119, 6912, 4830, 2527, 3724, 55, 9953, 5695, 4982, 7038, 8060, 3896, 7440, 1216, 342, 785, 2054, 8199, 2616, 5687, 4165, 9482, 826, 7549, 8496, 1333, 2331, 2645, 3088, 681, 2950, 5172, 1883, 1699, 9617, 2337, 1576, 8049, 1592, 1406, 6722, 1251, 6217, 9498, 8283, 7838, 8609, 549, 2371, 854, 3143, 8550, 4809, 8781, 3351, 3434, 1217, 5319, 7922, 6766, 1463, 353, 659, 4671, 3715, 2113, 1029, 6671, 9886, 8735, 1268, 4219, 3059, 6276, 7490, 6778, 393, 2193, 2907, 5593, 1185, 7332, 1498, 2805, 9403, 6935, 9517, 5625, 3325, 698, 8983, 441, 7734, 5586, 9476, 3629, 9440, 641, 1490, 9218, 6853, 4631, 687, 3821, 2500, 2025, 2029, 9634, 899, 2786, 381, 980, 3639, 5928, 3755, 126, 2194, 7696, 102, 4449, 2028, 8500, 1235, 239, 5925, 2500, 306, 8599, 3754, 3083, 6257}; int k = 4098; cout << sumOfMaxSubarray(nums, k); </pre>			
✓ <pre> vector<int> nums {7238, 2419, 2463, 3950, 955, 8955, 1124, 1768, 721, 7816, 1856, 288, 6749, 2714, 7121, 2615, 8454, 3441, 7864, 8124, 9173, 4143, 462, 1464, 1083, 5840, 4963, 6790, 1528, 1319, 3416, 9703, 6824, 8475, 3966, 8996, 3440, 6370, 590, 85, 1802, 5180, 2906, 8819, 7709, 7075, 6099, 4953, 1713, 6241, 2013, 1680, 6785, 819, 5280, 2619, 9281, 8349, 7249, 7760, 2274, 2238, 2192, 5928, 9652, 7797, 9011, 2936, 3452, 1348, 4611, 2993, 7110, 1396, 8325, 5926, 1117, 1766, 795, 9484, 9288, 7864, 972, 806, 3223, 6727, 5442, 2017, 7615, 6931, 7410, 6189, 145, 7111, 3570, 8638, 3554, 6155, 7862, 402, 822, 7051, 203, 404, 5103, 986, 1047, 6080, 7909, 5383, 6912, 5592, 3990, 4239, 6845, 3542, 5328, 1683, 7994, 1171, 6971, 9883, 4912, 9111, 3933, 3914, 2436, 2434, 4617, 2404, 9999, 5610, 24, 1646, 4446, 8198, 794, 4325, 7066, 1643, 7502, 9122, 7450, 8628, 7952, 989, 9256, 5816, 6275, 6047, 2818, 3095, 3728, 732, 8434, 1444, 2605, 5008, 8208, 6186, 2721, 6169, 6885, 664, 1239, 9382, 2788, 6526, 226, 9790, 517, 93, 8416, 5556, 6545, 5450, 8801, 7898, 5973, 5907, 6649, 724, 5392, 9541, 820, 1630, 3103, 3230, 7002, 528, 8668, 4734, 3236, 2929, 3667, 19, 990, 586, 741, 8741, 460, 4380, 7519, 8495, 3810, 2324, 5629, 1648, 2415, 4446, 5052, 7217, 7210, 4393, 4030, 3767, 3904, 2729, 6033, 4091, 5372, 8386, 7202, 4424, 8088, 628, 7470, 8788, 5970, 6130, 2185, 5235, 6819, 327, 642, 7337, 7020, 8683, 1735, 1551, 6770, 7042, 9925, 5496, 589, 4582, 9875, 7763, 731, 2806, 8015, 4307, 3370, 8111, 7257, 334, 229, 6906, 1871, 6131, 286, 1894, 5528, 9600, 2877, 8488, 4437, 3923, 3256, 6128, 7493, 991, 7028, 9021, 4312, 4385, 9594, 9363, 2438, 9491, 4542, 332, 1825, 6512, 8524, 4238, 5112, 9049, 868, 796, 6894, 7400, 7144, 2607, 314, 2468, 4306, 2182, 9904, 676, 668, 2235, 8368, 71, 7674, 7880, 77, 2643, 6138, 121, 2993, 3949, 4064, 3476, 6687, 5689, 1307, 7383, 9881, 3121, 2236, 8257, 83, 1995, 5464, 9076, 6152, 4902, 3964, 1657, 1723, 3446, 976, 7892, 7120, 6561, 1455, 3577, 6605, 8616, 2309, 3247, 2251, 2106, 7354, 1797, 1041, 4709, 1295, 5939, 9043, 9374, 4822, 481, 4359, 2739, 3271, 1581, 1016, 7776, 6800, 4089, 4862, 9359, 4381, 4532, 7646, 8035, 2136, 234, 4190, 1302, 371, 6750, 9916, 3665, 723, 4551, 5311, 8797, 5471, 4376, 142, 600, 1210, 53, 160, 9274, 8765, 4787, 4644, 2392, 488, 815, 2433, 1736, 2954, 9666, 6672, 8038, 48, 1667, 7881, 9287, 1690, 6413, 6828, 7068, 1508, 3458, 834, 1065, 8228, 7713, 786, 1094, 8399, 3716, 4375, 4662, 9249, 1133, 6210, 7095, 9260, 2866, 6765, 3481, 640, 5814, 8224, 8850, 2701, 314, 6407, 1809, 5045, 1332, 6659, 5025, 2533, 8863, 3696, 9745, 1365, 4275, 2487, 3890, 1022, 678, 341, 1552, 2204, 3603, 3884, 3428, 5604, 970, 2168, 9849, 8791, 4345, 4693, 7023, 5239, 2153, 479, 460, 6618, 8379, 3145, 3449, 8717, 6341, 6434, 9884, 1323, 5704, 5749, 3522, 894, 7076, 609, 7712, 8053, 8494, 6968, 7209, 3031, 16, 2171, 6754, 1702, 4402, 1302, 5076, 2759, 712, 4488, 6418, 7234, 3275, 5622, 8060, 4933, 2861, 8033, 7187, 9181, 8647, 4980, 495, 6392, 5973, 2482, 3261, 8021, 5413, 7600, 2988, 8845, 1819, 6751, 2458, 6, 8441, 6109, 4266, 1381, 3053, 7580, 1514, 5124, 2283, 1795, 2242, 2612, 2317, 6691, 5247, 661, 1368, 7421, 2352, 2451, 1825, 8642, 6208, 4579, 7280, 747, 4054, 8607, 5124, 2330, 4810, 4510, 4998, 1, 7353, 3784, 4866, 9810, 2699, 5928, 9444, 1126, 3650, 4211, 3839, 5361, 7238, 8888, 2676, 7606, 7071, 3960, 4321, 9288, 2450, 3768, 2024, 1441, 5199, 6133, 5601, 4550, 5082, 6199, 386, 2234, 8601, 905, 1534, 8753, 3742, 7160, 2108, 6984, 6982, 3209, 2410, 5140, 1456, 7168, 3618, 1208, 9666, 5135, 7308, 3836, 4697, 7340, 9187, 3082, 9951, 1798, 1388, 1033, 4130, 2393, 1460, 6592, 8242, 4703, 2602, 2899, 4603, 969, 9245, 2802, 4596, 1343, 8247, 1875, 4087, 677, 2394, 8428, 8354, 8447, 1562, 8149, 3549, 3382, 6092, 1902, 7806, 1467, 2012, 8621, 7929, 3355, 1157, 5227, 5244, 8163, 2845, 2719, 9290, 5222, 6527, 2074, 8661, 1373, 2957, 7861, 897, 9230, 6677, 3368, 8391, 4625, 2019, 1773, 264, 8462, 600, 4197, 5299 ...snip... 5656, 3856, 5100, 2655, 6245, 2754, 3468, 5041, 2715, 608, 4759, 701, 1628, 4400, 4845, 1175, 3899, 7323, 1829, 2104, 2974, 302, 8116, 1121, 562, 8627, 2528, 1759, 2625, 6207, 8181, 8207, 1568, 2637, 6695, 1412, 6460, 3883, 6602, 5639, 4086, 3652, 1423, 1087, 6859, 1614, 2703, 1665, 9785, 8932, 5612, 7369, 6143, 4502, 7302, 9914, 9989, 4048, 7964, 1954, 492, 4856, 2004, 1058, 1751, 3388, 3117, 7500, 8810, 4524, 4514, 3037, 578, 3699, 8274, 526, 6533, 700, 6574, 1990, 9163, 4483, 2626, 9922, 8221, </pre>	23355459	23355459	✓

Test	Expected	Got	
<pre> 3689, 9783, 2359, 7229, 2492, 7704, 6342, 5095, 7063, 4500, 482, 5686, 1397, 3300, 8911, 4745, 2988, 462, 1195, 9575, 2327, 2169, 33, 1330, 5044, 7265, 6843, 9976, 5568, 2445, 9224, 2847, 6208, 415, 6543, 8978, 9297, 2236, 5491, 2919, 9858, 5533, 8919, 2548, 2864, 2997, 8276, 1793, 7650, 6436, 9560, 7086, 5863, 6156, 6961, 8008, 8093, 2626, 875, 7097, 8531, 9766, 6614, 5902, 1537, 2790, 6043, 1629, 710, 6096, 7376, 998, 3106, 935, 7066, 137, 5974, 5848, 5635, 5256, 5256, 3166, 1373, 4350, 4315, 98, 9910, 3873, 6127, 6471, 828, 2324, 6164, 1637, 834, 4881, 917, 308, 3763, 8501, 3267, 7244, 4614, 1371, 2271, 5274, 9411, 1485, 893, 7845, 1206, 6520, 8149, 114, 3665, 6768, 9675, 8981, 8279, 4004, 8257, 4121, 8934, 1619, 476, 8530, 5268, 5056, 6724, 4032, 5326, 7074, 4416, 3223, 7228, 7694, 2882, 1853, 3161, 6553, 1470, 100, 1538, 4744, 3201, 1017, 6189, 9435, 7615, 8443, 7180, 2330, 1965, 2400, 8450, 3406, 7153, 7732, 7941, 2243, 1399, 9300, 149, 4524, 821, 2234, 9883, 1400, 1601, 2713, 389, 3924, 769, 8844, 6130, 7937, 8420, 5578, 8025, 8839, 5791, 941, 6367, 554, 4794, 1331, 888, 268, 9404, 8884, 9821, 8339, 8516, 5718, 4207, 4956, 1130, 6525, 4651, 7232, 4086, 9702, 351, 182, 8303, 8198, 9116, 1882, 1878, 5007, 3740, 3932, 1076, 2355, 7461, 6460, 7120, 748, 9855, 4485, 8582, 5051, 5443, 3015, 645, 3327, 8965, 9401, 440, 7304, 5475, 3246, 5431, 9564, 1630, 536, 1102, 1866, 2040, 139, 15, 6478, 4462, 595, 9556, 5381, 2265, 5608, 4919, 8145, 3376, 2497, 789, 8878, 592, 4791, 7024, 1974, 1374, 5819, 7525, 7059, 9244, 5275, 5262, 6083, 3834, 5425, 1576, 8672, 5551, 7884, 3799, 1440, 6808, 1635, 8232, 602, 1681, 9639, 1463, 827, 4338, 7673, 1347, 8136, 1851, 7196, 4191, 1941, 3251, 9099, 8682, 8933, 1949, 5230, 6033, 7314, 9292, 6681, 9652, 8096, 8777, 5302, 5334, 7309, 8418, 1253, 8856, 3724, 5093, 7968, 3929, 2066, 1118, 3076, 9255, 2944, 1517, 9836, 4281, 7778, 6363, 1056, 1120, 3959, 6991, 5567, 8570, 1747, 2428, 3065, 1705, 9201, 1645, 8656, 6788, 5773, 723, 8879, 5165, 5569, 5454, 6092, 6787, 1102, 347, 8698, 5496, 4077, 6525, 2866, 9494, 5542, 3503, 3203, 4029, 2753, 713, 2055, 3760, 4908, 385, 3032, 3329, 3404, 2563, 1986, 3162, 4487, 8948, 8107, 9037, 9314, 8660, 49, 4574, 1446, 2450, 1459, 7504, 1896, 5684, 2515, 397, 7257, 6791, 1210, 3051, 1051, 5796, 3773, 5323, 5526, 1332, 4734, 3795, 5546, 9045, 949, 7822, 2613, 1037, 6689, 2642, 6023, 2478, 9193, 195, 5347, 1643, 9672, 4694, 4649, 746, 25, 4856, 9219, 1066, 447, 5808, 2477, 5095, 345, 5347, 9368, 2764, 4947, 7127, 4512, 8777, 672, 1433, 5277, 1493, 4839, 3523, 9111, 2085, 3420, 2152, 6558, 397, 9648, 2930, 168, 9798, 2715, 1870, 8156, 7729, 441, 3606, 238, 6107, 1939, 448, 2636, 4898, 5680, 1029, 9849, 8551, 7900, 7492, 4648, 3076, 6935, 9511, 4348, 64, 8704, 6247, 4797, 6672, 9092, 1756, 9709, 3873, 6754, 7735, 3371, 3667, 417, 2500, 8200, 6504, 2036, 1146, 4104, 1741, 5868, 5189, 8988, 8104, 9919, 630, 6094, 5107, 4280, 2120, 8156, 6408, 3263, 3878, 1288, 526, 9867, 8236, 8575, 8642, 7781, 4347, 1833, 9561, 3758, 9458, 2577, 5829, 7427, 2489, 7445, 8326, 3072, 5743, 3408, 1811, 9027, 5542, 3718, 9626, 7875, 9309, 9312, 212, 2465, 2047, 8761, 4732, 3533, 7578, 5495, 4418, 284, 6435, 318, 7115, 9080, 3431, 9170, 8267, 5449, 661, 8907, 7941, 5619, 5275, 3550, 8689, 1903, 4222, 1434, 5232, 111, 8254, 8944, 3517, 1148, 585, 3142, 5861, 4114, 5278, 1549, 1688, 4831, 1557, 6034, 1824, 856, 2589, 9970, 8177, 2284, 6374, 5170}; int k = 7665; cout << sumOfMaxSubarray(nums, k); </pre>			
✓ <pre> vector<int> nums {4925, 1121, 1134, 9626, 7468, 9797, 8048, 1550, 1380, 8115, 3299, 235, 234, 2155, 168, 8735, 7027, 2208, 7401, 45, 4231, 5642, 461, 1904, 8789, 4315, 8251, 4560, 3590, 1477, 7186, 5683, 406, 1529, 1304, 1221, 721, 2690, 4945, 3691, 5162, 1732, 6515, 1685, 6722, 2255, 6007, 1970, 4615, 2297, 7139, 6488, 5675, 9146, 9350, 7668, 6743, 2441, 707, 8287, 6639, 648, 7410, 882, 2871, 6732, 8787, 6411, 7206, 7231, 3941, 3328, 2938, 6697, 2310, 6595, 2621, 6444, 1099, 7975, 6151, 2079, 1666, 962, 5827, 6430, 1654, 9380, 8278, 7247, 9156, 407, 2013, 8632, 1499, 3655, 55, 8944, 6201, 1075, 8548, 6739, 6660, 3157, 347, 1514, 7829, 5567, 1554, 1418, 1852, 1878, 3369, 8379, 2544, 2515, 2968, 3612, 2294, 6631, 8347, 3766, 4996, 2143, 5306, 238, 9382, 2329, 4926, 1294, 8512, 198, 4145, 8438, 2389, 1361, 7190, 6671, 4264, 2142, 2153, 3937, 4481, 8134, 2167, 1702, 915, 4834, 3153, 3486, 8763, 4502, 4365, 708, 7515, 8288, 8190, 6944, 9720, 6337, 3541, 1663, 1345, 1708, 9480, 2384, 7457, 5452, 5730, 1872, 3463, 3435, 3594, 825, 100, 3920, 2601, 1766, 5812, 4552, 8518, 602, 8992, 1864, 676, 3234, 8355, 9346, 5647, 6456, 2684, 1671, 2723, 4377, 4372, 5404, 5038, 3161, 9697, 2895, 2391, 6038, 2393, 7804, 3902, 7274, 2395, 5739, 1653, 277, 1386, 172, 6208, 7882, 9644, 8614, 5785, 1256, 4274, 460, 3207, 5917, 4409, 9810, 688, 214, 689, 5726, 5693, 2026, 138, 3250, 525, 1767, 376, 9945, 8876, 8540, 5487, 5584, 4309, 7082, 4143, 2071, 4982, 1077, 1654, 2904, 7891, 5599, 2690, 1415, 9710, 1462, 5302, 3258, 9813, 9165, 5743, 59, 5788, 1993, 4087, 8796, 413, 3645, 4618, 1963, 7629, 2712, 535, 9238, 8640, 2316, 5705, 6336, 5840, 1539, 1927, 6677, 2122, 4605, 7833, 3866, 4538, 5901, 8972, 8313, 4937, 3205, 7335, 2123, 8515, 4447, 2815, 8374, 9262, 8238, 9144, 602, 2583, 9012, 1901, 7974, 5183, 7029, 5586, 5392, 1399, 3451, 1639, 2901, 535, 300, 1824, 5182, 389, 6569, 6067, 2292, 5060, 7992, 1153, 8581, 379, 6373, 4633, 6312, 6744, 3028, 4170, 2781, 9476, 8425, 5447, 2392, 6937, 4252, 2880, 1839, 6513, 1063, 702, 2802, 2113, 8386, 1902, 1835, 502, 9170, 3203, 8006, 6111, 1165, 8005, 738, 2891, 7585, 8286, 3496, 1156, 7477, 3959, 61, 9052, 5840, 4078, 1347, 8294, 3465, 4571, 8452, 425, 4222, 2227, 1393, 2768, 9321, 7661, 4936, 4198, 6704, 5731, 7588, 6480, 6812, 8283, 1770, 1947, 5196, 9558, 9073, 6025, 267, 6201, 5873, 901, 671, 8362, 6416, 575, 6132, 2765, 1576, 1844, 5608, 6368, 1949, 5341, 8, 6579, 3005, 8572, 539, 508, </pre>	133466652	133466652	✓

Test	Expected	Got	
<pre> 5897, 3186, 5469, 266, 7796, 2916, 4687, 7433, 441, 6271, 9293, 5398, 6594, 5161, 704, 5907, 1270, 4794, 7397, 1916, 8802, 3868, 3501, 2011, 1531, 2367, 9335, 8489, 7507, 6800, 7643, 3127, 7115, 8474, 2424, 124, 1272, 13, 9905, 7935, 4329, 6510, 6131, 6711, 7703, 8048, 2090, 5824, 5070, 6130, 1620, 4370, 4059, 9064, 9014, 1245, 2694, 542, 9870, 1111, 3474, 9834, 2280, 7914, 5379, 5986, 6645, 3607, 3135, 7998, 6457, 287, 7416, 8287, 7108, 420, 4857, 8016, 9972, 7659, 767, 2220, 5145, 4821, 5914, 4384, 553, 5966, 84, 3491, 628, 1749, 1649, 4707, 7083, 3469, 9688, 1825, 6316, 4356, 657, 7712, 1089, 5218, 2943, 73, 2234, 3620, 5584, 5139, 8936, 762, 6802, 4866, 7996, 317, 8648, 9021, 3199, 1270, 5533, 4532, 5379, 5595, 1881, 8994, 4746, 1594, 6226, 719, 6070, 1131, 5749, 2234, 6144, 7728, 7941, 7269, 3500, 6833, 355, 7264, 5325, 454, 8174, 27, 4950, 1332, 109, 1014, 1025, 988, 2971, 3916, 4338, 2960, 6961, 3583, 3566, 1127, 507, 2629, 5289, 9507, 6036, 1796, 8494, 6017, 4931, 4970, 6579, 161, 8746, 8320, 6619, 791, 8140, 6127, 1693, 9617, 3856, 6180, 9237, 457, 1811, 8816, 1402, 6509, 1554, 9907, 4672, 1342, 6296, 4442, 1502, 1254, 8245, 566, 8915, 9789, 224, 9664, 1893, 1597, 511, 7242, 362, 1797, 980, 5751, 1434, 3115, 2553, 9181, 8644, 9285, 9727, 570, 7985, 9562, 4964, 2709, 2022, 1317, 1156, 7207, 2526, 7722, 3745, 2633, 2711, 251, 1157, 303, 779, 1485, 7271, 1814, 4162, 3303, 4765, 8410, 2758, 2123, 1809, 6923, 4583, 280, 951, 209, 6416, 9487, 6670, 4509, 5178, 565, 901, 8898, 7087, 2616, 4079, 4324, 3554, 117, 6090 ...snip..., 1227, 5451, 1845, 6120, 4398, 8668, 7231, 8373, 5688, 3130, 2029, 6030, 1827, 6058, 885, 7537, 6085, 2659, 1182, 2570, 3913, 5238, 7667, 3994, 2687, 5509, 6055, 4935, 6450, 2541, 5752, 7700, 3083, 9589, 2246, 7090, 8628, 6508, 9477, 4838, 672, 7950, 8274, 3261, 2729, 2824, 9923, 8238, 6010, 1180, 514, 1135, 2207, 9271, 6810, 4427, 1801, 2609, 7602, 1074, 1270, 6708, 1739, 7329, 9664, 6129, 805, 899, 84, 1599, 1047, 120, 551, 4053, 4462, 8838, 9156, 7946, 8933, 1581, 4169, 4827, 4998, 2031, 6115, 1405, 1280, 8912, 5894, 8452, 7418, 4906, 4898, 483, 9735, 6740, 1618, 1317, 8042, 340, 6887, 6075, 3723, 2595, 84, 9814, 1962, 688, 2747, 9784, 7188, 7943, 393, 5062, 2901, 6515, 5597, 5489, 3261, 8017, 5294, 246, 5461, 6440, 8270, 322, 8563, 9350, 729, 4601, 1036, 9596, 835, 7193, 8394, 9485, 2401, 2347, 4862, 899, 2662, 7354, 8600, 5475, 4577, 2086, 7349, 7234, 7061, 2187, 3415, 7319, 3162, 2238, 2715, 2610, 466, 7800, 4456, 814, 7774, 6935, 660, 5316, 8481, 7329, 1659, 256, 693, 7498, 6578, 1255, 4509, 2023, 4488, 1177, 4987, 2067, 4190, 9300, 3363, 5587, 3209, 8473, 2720, 4646, 9494, 51, 3622, 4079, 6589, 3940, 2967, 2469, 3273, 1074, 1753, 1210, 3687, 3219, 984, 9851, 3244, 469, 3112, 3048, 812, 7058, 2718, 981, 297, 4753, 4880, 6592, 2585, 3679, 25, 9188, 6059, 6784, 5250, 8205, 3243, 1009, 8285, 7933, 6260, 4608, 8404, 6990, 3125, 9999, 4898, 5496, 7559, 4180, 9743, 2037, 6478, 3972, 6162, 5236, 9102, 2562, 130, 3995, 592, 8450, 5038, 3553, 2004, 3299, 382, 4347, 105, 3148, 1305, 7717, 796, 3350, 8792, 4099, 4958, 9995, 4123, 2024, 473, 2189, 2122, 182, 3955, 3322, 4847, 2743, 9524, 5854, 2387, 4477, 6453, 1609, 9477, 9701, 9641, 4243, 4962, 4561, 7505, 2029, 737, 485, 1850, 1658, 799, 5394, 7113, 5919, 1483, 2994, 3963, 6758, 2722, 4867, 1430, 7049, 5597, 152, 1508, 221, 1369, 1532, 8497, 9756, 664, 8328, 3378, 6728, 9686, 7718, 5944, 9869, 9823, 9175, 7110, 3590, 5983, 8851, 6623, 1027, 1562, 703, 3796, 1546, 4925, 7524, 6189, 411, 3604, 1149, 1557, 2648, 3246, 4848, 5227, 5572, 4286, 6097, 7574, 6491, 8931, 580, 568, 2195, 7402, 7154, 823, 1826, 658, 2029, 3464, 192, 5616, 1719, 1228, 2670, 9636, 7472, 1232, 9207, 4124, 9855, 8896, 9814, 6168, 5395, 5324, 2280, 3326, 3946, 3149, 4499, 1823, 561, 5625, 615, 7913, 4829, 5493, 7044, 4814, 378, 6715, 4989, 3893, 1346, 4077, 412, 5904, 4688, 3128, 2549, 782, 4661, 9308, 8095, 7505, 331, 9339, 9184, 1906, 1545, 4816, 3112, 5636, 4662, 1874, 1746, 8630, 8577, 185, 8550, 7661, 1696, 8401, 1378, 3314, 855, 1181, 1979, 295, 3196, 6925, 94, 1728, 2697, 9602, 8027, 5536, 865, 9697, 597, 1441, 7303, 7, 4387, 7928, 9467, 8748, 1051, 6888, 3510, 6683, 5343, 4854, 7208, 2417, 5864, 8093, 5792, 4100, 7043, 291, 444, 1238, 345, 3483, 2593, 4842, 7075, 6061, 7165, 8998, 6910, 3411, 8363, 2713, 75, 4405, 7752, 5119, 1770, 8391, 1780, 3330, 8888, 4033, 9418, 7300, 2726, 2511, 4899, 3723, 3901, 891, 6231, 396, 4046, 936, 1718, 3883, 2399, 8560, 7911, 17, 4595, 5968, 3375, 5183, 3320, 8289, 6340, 1298, 455, 1675, 5970, 6727, 2040, 9286, 1486, 8993, 6789, 1985, 6729, 1961, 3311, 5238, 9943, 7830, 3558, 6269, 3408, 1069, 8085, 1729, 7975, 1866, 2364, 6256, 9721, 6224, 6963, 950, 2712, 1341, 4203, 2572, 5705, 34, 4583, 3917, 7227, 849, 4491, 1994, 266, 3447, 5732, 1230, 4386, 9917, 6248, 7268, 5271, 74, 2383, 712, 1812, 9371, 3314, 2555, 3523, 7365, 6307, 4043, 6701, 1177, 8957, 818, 1042, 6362, 5998, 446, 7275, 860, 7282, 8081, 8765, 4909, 241, 442, 1953, 2136, 5564, 1780, 883, 55, 468, 5405, 1767, 6457, 2780, 5977, 2403, 6693, 2512, 136, 6025, 5838, 2902, 7909, 4611, 919, 2691, 7841, 7380, 4425, 9921, 1392, 1452, 3026, 1152, 1970, 7388, 7376, 2731, 3252, 859, 1199, 9980, 9384, 9768, 4463, 2179, 7698, 5489, 3321, 7507, 8245, 756, 8835, 5837, 9246, 6708, 1635, 5232, 4286, 3951, 1584, 481, 1438, 4589, 6575, 6936, 5822, 3323, 1469, 588, 3502, 6248, 4712, 2720, 9905, 8399, 2131, 9005, 31, 5111, 4617, 9074, 2138, 3320, 17, 2570, 4755, 4671, 5763}; int k = 6653; cout << sumOfMaxSubarray(nums, k); </pre>			
✓	vector<int> nums {4413, 211, 4251, 7577, 2258, 4873, 4283, 226, 1222, 277, 6582, 1447, 8873, 5138, 2826, 4647, 4512, 2596, 2845, 2996, 7046, 636, 8972, 7464, 2031, 4027, 9408, 6757, 5501, 4570, 3281, 4145, 7607, 42, 47, 369, 1798, 6040, 4207, 1653, 3842, 1123, 5059, 2005, 6845, 8817, 1566, 4186, 8894, 9646, 166, 5504, 6720, 7399, 5803, 9850, 4588, 5542, 1670, 7063, 2613,	400233382	400233382 ✓

Test	Expected	Got	
5679, 7957, 1828, 6930, 9674, 2935, 9634, 9761, 7794, 2659, 2169, 4106, 5549, 3886, 1196, 4962, 77, 5212, 3919, 721, 1959, 3990, 5655, 4326, 6705, 8317, 338, 2921, 1185, 8878, 8231, 7281, 7076, 7351, 3518, 1895, 461, 7049, 5618, 9755, 1342, 1438, 6262, 8886, 9435, 220, 9575, 3600, 8749, 9067, 3360, 6882, 2597, 9924, 5891, 3502, 9666, 747, 7191, 7662, 1092, 7742, 8317, 9067, 1042, 4510, 4424, 1231, 8873, 3933, 188, 2684, 3861, 4440, 4277, 8734, 8245, 3096, 7163, 5967, 5098, 9460, 1717, 3482, 1441, 2681, 9196, 5186, 7826, 44, 1137, 7587, 519, 2383, 5922, 361, 1360, 9042, 2217, 1185, 3734, 7004, 213, 2284, 6799, 5515, 9021, 6187, 2563, 8952, 1138, 871, 892, 3742, 2356, 3072, 6259, 9916, 4632, 1449, 4232, 5257, 7065, 6225, 913, 8792, 189, 6389, 552, 8812, 1497, 3380, 1758, 9854, 1415, 660, 340, 7836, 230, 5763, 8092, 6100, 8147, 2720, 2171, 290, 1111, 7419, 2706, 8390, 1809, 6122, 6205, 921, 9558, 782, 2236, 3138, 6342, 2438, 9872, 2708, 1025, 4194, 2008, 7946, 7289, 9978, 636, 70, 1801, 128, 121, 5797, 6454, 3218, 943, 1271, 9670, 7509, 6023, 5038, 8111, 2479, 7034, 2597, 1824, 3220, 7870, 6228, 7381, 2164, 4849, 641, 9255, 9218, 1435, 6920, 4643, 5239, 879, 1902, 93, 5000, 130, 1171, 9332, 2438, 7951, 4342, 487, 8005, 9339, 8669, 9768, 15, 8741, 4101, 7840, 9177, 6319, 6279, 2387, 9286, 4211, 780, 1574, 491, 6576, 7058, 6366, 9812, 7460, 6815, 5626, 8709, 3647, 2757, 8900, 514, 8213, 951, 4644, 8323, 5364, 1035, 8022, 7666, 4854, 8576, 205, 3511, 1209, 3378, 984, 5894, 3726, 620, 7523, 2039, 1855, 510, 7585, 1591, 5508, 2079, 3081, 2637, 4087, 196, 2286, 3184, 9943, 3569, 5684, 4326, 234, 223, 5728, 3360, 7639, 1887, 5679, 293, 5747, 7790, 6106, 8603, 5129, 7610, 1880, 8797, 9583, 9604, 7812, 2378, 7191, 4918, 9601, 3546, 2756, 4375, 2699, 1782, 5630, 8460, 5776, 1096, 9671, 34, 751, 1697, 3630, 921, 6255, 42, 1028, 1651, 8467, 7933, 6126, 8525, 5276, 453, 8890, 1309, 8800, 8499, 5109, 4653, 6490, 5715, 540, 4215, 6658, 2174, 6498, 4610, 5087, 7392, 1745, 6139, 1497, 1805, 9796, 1895, 8084, 5264, 3269, 40, 7626, 8932, 8691, 5630, 6344, 5911, 9558, 0, 1148, 4652, 1670, 8860, 1981, 9872, 6216, 9883, 1227, 6894, 5274, 328, 5177, 6727, 5186, 3315, 7993, 8412, 2914, 977, 4246, 4910, 3806, 4374, 919, 7785, 3537, 4981, 2339, 4987, 3724, 4695, 3232, 8363, 2481, 602, 4001, 3192, 3680, 5116, 6783, 3008, 656, 9417, 4534, 6291, 8304, 4817, 2500, 9078, 189, 6430, 1981, 7148, 6327, 9012, 6227, 5202, 2880, 3657, 2981, 7786, 1412, 9047, 9509, 9994, 9119, 2272, 4638, 2069, 9180, 9362, 4245, 411, 74, 2235, 8575, 2493, 4532, 3006, 3364, 6964, 5872, 6470, 994, 3826, 2407, 2955, 277, 3032, 7671, 2531, 2150, 9786, 9462, 4122, 2210, 2736, 2968, 6213, 5676, 2110, 4083, 1172, 5077, 7367, 769, 1862, 2411, 6493, 5189, 2113, 2240, 828, 177, 8642, 2380, 4312, 5976, 9842, 4806, 5956, 2064, 2426, 6191, 3988, 906, 1160, 2415, 2613, 1380, 9634, 5828, 5371, 7646, 2042, 9592, 2204, 4877, 707, 8082, 7596, 3949, 2083, 4096, 3959, 4826, 8746, 2468, 1654, 3312, 7954, 396, 5948, 17, 9031, 71, 9840, 2823, 4968, 676, 1249, 2139, 9695, 733, 8661, 8709, 7038, 3109, 6082, 7109, 9434, 1789, 2624, 9602, 1964, 1677, 782, 273, 8069, 6209, 2580, 1352, 5746, 5409, 6121, 1459, 753, 1222, 3239, 2950, 1603, 7466, 332, 8372, 7347, 192, 6930, 4000, 2469, 9796, 8529, 2611, 6962, 2195, 9314, 6061, 979, 1038, 8551, 2943, 9996, 1019, 8570, 9384, 3548, 8749, 6884, 9518, 7025, 4005, 9295, 4143, 3457, 9960, 2686, 873, 4566, 3699, 9883, 5367, 986, 6061, 6369, 797, 3236, 6588, 155, 2358, 4689, 4315, 7356, 9067, 7862, 1242, 7553, 6279, 1861, 5398, 4244, 9897, 1071, 2475, 2338, 8842, 329, 6951, 8431, 7472, 9889, 448, 9208 ...snip... 51, 5195, 1586, 1415, 1260, 5389, 1707, 4564, 4257, 2466, 4422, 1363, 9297, 4228, 3974, 9132, 44, 6455, 7859, 38, 263, 5737, 2414, 6961, 6947, 1037, 6706, 1850, 7196, 296, 1260, 7706, 3587, 811, 8092, 4496, 5454, 7106, 2603, 8593, 1438, 1452, 5775, 9261, 2881, 941, 8462, 9651, 650, 5632, 1936, 728, 670, 1073, 5063, 8980, 8271, 3258, 1363, 5277, 9566, 6220, 785, 3400, 2269, 48, 8676, 8989, 1699, 9966, 5415, 7451, 9635, 7038, 1778, 1564, 8815, 2830, 123, 1162, 8601, 3508, 2798, 2418, 2923, 3107, 8752, 7240, 472, 211, 4833, 5563, 3794, 2668, 8304, 3642, 28, 4339, 2147, 1127, 563, 5993, 6727, 9579, 7257, 4960, 645, 122, 5282, 4160, 5430, 6280, 1475, 9385, 728, 765, 7623, 1001, 1424, 2261, 2948, 5728, 3187, 2059, 5550, 1890, 5264, 6743, 3514, 3628, 257, 2149, 9910, 8715, 4146, 7576, 1927, 1610, 2464, 4558, 7138, 2394, 806, 7318, 4991, 7687, 7875, 896, 8203, 8185, 7584, 4260, 6013, 456, 6065, 7429, 571, 2100, 4139, 6247, 1358, 1283, 3861, 2143, 8740, 7628, 6305, 970, 2180, 7427, 268, 2273, 3767, 4806, 4772, 6283, 5157, 7960, 6570, 6787, 3693, 5756, 9992, 2076, 7498, 165, 8821, 9346, 1890, 1323, 1190, 9859, 336, 3024, 1223, 1741, 3349, 2035, 2859, 1754, 6842, 9045, 657, 2282, 2921, 9098, 5799, 9343, 6125, 2099, 6547, 8495, 423, 1850, 5199, 5317, 6146, 4045, 5909, 7097, 6442, 8014, 5403, 2209, 280, 2458, 785, 9352, 8401, 6137, 7257, 3310, 112, 7142, 1478, 1377, 9648, 7248, 8920, 6826, 4031, 7119, 9414, 1083, 3087, 8874, 782, 6679, 9919, 5337, 1961, 4978, 1226, 9367, 5090, 4859, 6398, 10, 88, 9463, 3088, 9478, 8434, 1489, 3710, 3055, 6915, 1358, 7364, 7198, 2959, 4293, 6995, 693, 1420, 8463, 9215, 9670, 9804, 6859, 1528, 705, 6250, 6134, 540, 2380, 6819, 9065, 8650, 9528, 9623, 2643, 506, 9266, 6351, 760, 5804, 3240, 8562, 8410, 5086, 3932, 2227, 5927, 4049, 4541, 1757, 6788, 9166, 8580, 1890, 5176, 4597, 3628, 5455, 8716, 7047, 1719, 8181, 242, 8981, 4444, 8980, 1698, 1047, 6981, 296, 3597, 1010, 9272, 5416, 3724, 6356, 5636, 4829, 8377, 1746, 3807, 1488, 3824, 6132, 6989, 455, 8951, 4734, 8244, 8130, 9724, 1778, 9086, 2256, 734, 7477, 6078, 695, 3822, 3472, 5606, 9508, 3863, 5928, 2312, 1120, 4572, 8623, 9228, 9892, 2450, 4780, 3218, 3550, 7886, 8905, 6713, 8868, 8591, 3995, 1039, 3000, 1166, 295, 2153, 1453, 5349, 1747, 5333, 6409, 5679, 1067, 6862, 2556, 9583, 838, 1409, 3952, 3235, 6632, 8670, 2033, 1418, 8136, 8160, 3822, 7827, 6341, 6608, 9824, 5272, 9188, 3622, 3435, 6667, 3757, 5634, 6436, 288, 5848, 3771, 7938, 5548, 1026,			

Test	Expected	Got	
<pre> 5670, 3495, 1243, 5236, 3497, 5136, 3608, 7957, 1120, 7963, 9945, 2175, 3599, 57, 281, 128, 8729, 7044, 8794, 5266, 2721, 7443, 2090, 1692, 4671, 6124, 1446, 7514, 5302, 4505, 9485, 5067, 9523, 2942, 6262, 3370, 2328, 2233, 4595, 8544, 2474, 7115, 9624, 7543, 6856, 7415, 3495, 3090, 2173, 9736, 8654, 4699, 184, 439, 2148, 3692, 5914, 3230, 8879, 9235, 6191, 4439, 9626, 6048, 816, 4232, 6213, 3168, 7848, 2979, 1181, 1631, 5670, 7176, 7278, 5802, 8675, 5695, 726, 5347, 6469, 6435, 1275, 3640, 7442, 5248, 2096, 9414, 3049, 8721, 5108, 7177, 590, 9011, 3115, 6887, 4516, 30, 2370, 6239, 1280, 2517, 6125, 8828, 6843, 7932, 5587, 494, 6616, 4873, 1308, 7083, 8183, 5630, 2778, 225, 9353, 6566, 4966, 9648, 197, 7358, 6812, 2262, 5002, 4134, 5945, 570, 8790, 8488, 164, 1709, 7271, 632, 3803, 4574, 3632, 6348, 2463, 1613, 3358, 1262, 2355, 9618, 2619, 7904, 673, 3503, 7587, 6496, 5406, 2483, 2981, 4924, 7599, 3553, 1052, 7591, 4652, 5680, 543, 1549, 7491, 2388, 5590, 4515, 8064, 605, 5182, 4258, 2747, 8366, 5006, 9654, 9827, 1568, 1051, 9330, 7786, 40, 1705, 4550, 3073, 8839, 4067, 2692, 774, 7633, 1021, 4575, 6314, 1937, 8183, 9413, 3640, 298, 5589, 5999, 7139, 2560, 6605, 678, 2671, 9177, 8944, 1176, 1923, 2686, 9021, 1467, 5245, 7318, 9155, 4882, 934, 8005, 9317, 1262, 1031, 7536, 1235, 3222, 4093, 8053, 1622, 8841, 362, 3913, 4507, 8787, 1500, 3377, 4517, 9487, 3697, 8917, 8748, 7133, 1227, 4482, 810, 740, 7120, 336, 6968, 9049, 2501, 2099, 3332, 3953, 5533}; int k = 9973; cout << sumOfMaxSubarray(nums, k); </pre>			
✓ <pre> vector<int> nums {8678, 5235, 6156, 9412, 9508, 9313, 801, 3687, 4706, 9528, 7699, 6055, 7037, 1750, 5915, 4444, 7578, 3468, 5331, 5142, 3898, 757, 4093, 2237, 7298, 3254, 135, 539, 2216, 3413, 8365, 9176, 8224, 799, 9520, 5926, 6456, 6868, 7798, 6526, 896, 6742, 8369, 4730, 1023, 627, 1876, 6293, 945, 766, 2372, 4608, 5573, 1685, 3851, 919, 183, 1102, 1203, 3875, 3577, 5012, 7777, 7521, 1559, 6433, 1743, 5875, 4123, 5803, 6655, 8549, 2910, 6356, 1149, 834, 2065, 4765, 1061, 9752, 7787, 9787, 6564, 1194, 6323, 253, 8243, 3493, 784, 844, 9153, 2725, 3759, 675, 1712, 2136, 8261, 8834, 932, 7603, 4597, 6974, 1303, 2324, 8468, 8939, 2528, 1400, 4492, 3836, 9876, 8254, 8318, 3162, 5438, 1579, 8139, 372, 1580, 751, 1077, 2598, 7978, 7768, 7841, 3357, 7811, 6597, 8560, 9337, 8500, 7822, 7006, 3956, 8131, 9635, 4874, 2466, 4000, 1826, 4135, 1305, 8892, 5517, 1715, 3700, 1527, 6970, 1750, 819, 9566, 2701, 9229, 2196, 2213, 8051, 2408, 1496, 2336, 6465, 9505, 6687, 9177, 3308, 9214, 6885, 461, 4250, 88, 2322, 8680, 2041, 5564, 6124, 1138, 8942, 1202, 3312, 8985, 9713, 9389, 3089, 7601, 4103, 226, 1807, 5242, 5456, 7244, 7324, 9553, 978, 3263, 7035, 5466, 902, 5052, 4154, 5772, 3617, 1653, 2994, 1917, 7574, 4849, 3804, 1040, 9609, 658, 4087, 6971, 6679, 2021, 5239, 3550, 629, 4170, 6543, 2689, 4694, 1941, 6866, 2350, 5825, 2660, 8406, 8719, 8578, 9857, 7947, 1241, 7168, 2234, 1915, 7913, 5087, 6800, 8114, 8253, 583, 3549, 8288, 726, 8644, 9353, 8044, 1132, 9935, 9054, 3387, 4563, 541, 9291, 7170, 542, 4215, 102, 3464, 8065, 1610, 469, 5935, 8202, 8015, 5151, 1157, 1411, 5773, 2191, 6614, 1057, 9678, 3993, 4516, 6033, 912, 2592, 1609, 4836, 6125, 7990, 3964, 343, 8909, 821, 8003, 1945, 266, 5886, 584, 3175, 868, 2464, 2351, 3226, 4022, 540, 7431, 7699, 9193, 8839, 2351, 7429, 2856, 8463, 7082, 9220, 7155, 1681, 1526, 2237, 2519, 906, 2841, 1126, 238, 5100, 1230, 231, 84, 9892, 1711, 2569, 58, 7386, 2450, 8734, 6734, 6425, 2089, 5378, 2202, 7759, 2165, 5040, 827, 5783, 9403, 5463, 6645, 5301, 9946, 8911, 7190, 6689, 3691, 3015, 3336, 1128, 1434, 8769, 6219, 2656, 4500, 9776, 8482, 3819, 6977, 5152, 6909, 5949, 6661, 9127, 5559, 1473, 6689, 5722, 753, 2664, 2303, 2179, 3182, 868, 1742, 4594, 3584, 2741, 7594, 962, 9844, 2177, 9760, 3389, 7673, 4297, 5712, 4403, 8278, 3546, 3236, 1527, 7292, 145, 1776, 5866, 4554, 6771, 2043, 3772, 9008, 8172, 1245, 8399, 4471, 2838, 1113, 415, 5439, 2652, 269, 463, 7000, 5791, 2839, 8550, 8197, 5593, 2283, 9370, 8877, 9856, 6786, 6603, 1540, 4093, 3781, 4131, 2842, 878, 906, 4877, 8418, 6489, 7406, 8164, 2742, 6098, 4980, 5501, 169, 2367, 380, 4791, 2109, 7177, 3864, 3937, 2454, 6902, 3020, 5373, 3615, 309, 1444, 1589, 3595, 1833, 2585, 3003, 1375, 826, 4215, 1653, 3790, 6920, 2761, 4360, 1364, 9810, 2358, 7604, 1640, 182, 5697, 6065, 2540, 4121, 1515, 3400, 2630, 6473, 9675, 6421, 356, 2897, 353, 6316, 2779, 1453, 6583, 9337, 2241, 6847, 1161, 9180, 3612, 6657, 4999, 4885, 1002, 1868, 2216, 4173, 2930, 6744, 1393, 1413, 2331, 4837, 1031, 4946, 9052, 2138, 3105, 5370, 8100, 2651, 6609, 3080, 3163, 1696, 3269, 6070, 7592, 7908, 4729, 6699, 8356, 9708, 6447, 3715, 7918, 8905, 7314, 2351, 1873, 3359, 55, 4044, 1748, 4582, 372, 9048, 6800, 2384, 1574, 7314, 375, 3341, 2638, 1105, 802, 8124, 4779, 9449, 9644, 7251, 1694, 377, 7710, 736, 4074, 8136, 3978, 1825, 6971, 674, 9980, 6449, 6136, 477, 7725, 9044, 9596, 337, 170, 8053, 3521, 3431, 1845, 7469, 1886, 9312, 2666, 2487, 659, 730, 8981, 6850, 4321, 543, 5042, 2295, 9488, 5982, 3877, 4731, 3227, 142, 2443, 9255, 4271, 7776, 1728, 3718, 1476, 373, 7138, 2335, 6491, 3744, 7238, 3508, 521, 2477, 474, 3354, 3394, 17, 9019, 7418, 8666, 2756, 8472, 983, 1704, 8397, 9606, 3233, 7452, 5916, 9205, 5673, 6498, 6318, 9123, 7302, 9815, 7644, 432, 9079, 971, 7749, 4114, 6970, 4040, 8910, 9626, 4413, 9043, 727, 6006, 8403, 1808, 3968, 9800, 9593, 9914, 4299, 7063, 9977, 3731, 5619, 6660, 1950, 3961, 4254, 7040, 9994, 4971, 335, 3692, 1717, 6554, 9327, 5634, 6567, 6195 ...snip... 18, 6958, 7467, 9540, 7547, 4925, 9973, 7236, 2749, 3264, 4701, 2182, 5151, 1893, 2671, 6289, 2443, 9839, 5997, 9584, 3870, 3025, 4450, 5109, 2676, 4669, 835, 8527, 4948, 4925, 2167, 3384, 55, 3714, 6953, 8498, 4751, 9083, 9755, 5987, 3334, 1033, 3804, 2846, 6852, 1616, 5421, 756, 9014, 6624, 5201, 4875, 4363, 9205, 1252, 6750, 8720, 284, 7118, 7189, 206, 7032, 5383, 1910, 6834, 6563, 2365, 4284, 8718, </pre>	775142478	775142478	✓

Test	Expected	Got	
<pre>3976, 8099, 2279, 7759, 2353, 9357, 1780, 4735, 3836, 1577, 2319, 7275, 36, 2832, 8902, 8919, 861, 958, 5087, 6928, 4959, 189, 5277, 8882, 2551, 1657, 5215, 6348, 619, 400, 7770, 561, 4460, 5864, 2808, 1360, 7713, 6389, 2704, 1366, 1185, 5275, 152, 1730, 8593, 4289, 1961, 4217, 7215, 8551, 1666, 3628, 8974, 5705, 4015, 8887, 3796, 1172, 8172, 7707, 5040, 9625, 372, 1454, 6895, 6624, 3967, 8689, 4417, 5647, 5419, 1074, 1191, 2746, 4825, 5027, 3168, 9613, 6588, 4235, 1765, 40, 494, 1596, 2193, 1194, 3476, 9269, 117, 7905, 2579, 936, 2411, 8785, 1283, 8174, 9145, 515, 9074, 7978, 7656, 2512, 4732, 4859, 1190, 2943, 6602, 7334, 2521, 3483, 6936, 5780, 9509, 4152, 1436, 4520, 2367, 7127, 1508, 1907, 1299, 4088, 7166, 811, 7667, 4031, 8542, 3097, 7809, 8699, 2844, 6675, 7744, 1475, 8764, 8250, 21, 1377, 2674, 2500, 7763, 756, 2165, 4028, 3112, 8944, 2355, 6376, 9602, 313, 6501, 1515, 1398, 2976, 7589, 252, 8098, 3388, 733, 1432, 5457, 7821, 5218, 6980, 111, 2734, 4481, 9605, 6957, 1817, 9929, 3587, 5128, 5391, 4745, 1451, 581, 7959, 6971, 2052, 4656, 3271, 2929, 5636, 1509, 1617, 6430, 2468, 8878, 2288, 9753, 5681, 8927, 1092, 7534, 4773, 9774, 57, 1093, 5654, 8310, 1822, 6647, 5644, 8734, 1525, 5998, 1657, 358, 1302, 9772, 8420, 5106, 394, 8575, 4646, 394, 2549, 5971, 8809, 304, 6338, 2505, 2254, 3796, 9973, 3591, 2382, 2405, 8461, 9424, 5955, 6360, 1673, 3573, 7400, 2286, 3469, 9483, 14, 9617, 1797, 1103, 8800, 9183, 5177, 9000, 4521, 1083, 4865, 2652, 917, 2256, 7483, 277, 2085, 4375, 6955, 1349, 1351, 3259, 6530, 5304, 7366, 7659, 978, 5610, 9565, 4608, 1872, 4841, 1330, 9807, 1595, 3771, 2961, 7473, 2365, 3074, 1301, 8113, 5920, 2607, 5806, 5270, 3378, 1062, 1712, 6482, 8316, 5697, 7977, 1018, 5791, 647, 7877, 1958, 368, 126, 3765, 2411, 3094, 558, 9405, 8344, 1105, 4744, 5421, 142, 4733, 5751, 6627, 7524, 1319, 6134, 8014, 2233, 784, 2771, 3270, 4667, 8592, 5682, 9466, 1176, 6877, 676, 5546, 3283, 8731, 3114, 1911, 7529, 9830, 7364, 6407, 9684, 8990, 3455, 4381, 3551, 9074, 9759, 7737, 5788, 6001, 457, 5756, 5619, 9404, 20, 4495, 9451, 576, 5584, 5012, 1582, 3363, 5624, 1260, 7160, 3213, 5796, 6826, 1376, 7721, 2067, 105, 5946, 4939, 3495, 4490, 876, 212, 1137, 1936, 1952, 4852, 8030, 9232, 4090, 5279, 9042, 418, 5626, 6388, 1076, 253, 3508, 698, 7338, 8739, 9550, 4747, 791, 552, 8793, 7046, 4786, 3865, 9060, 5199, 2616, 7914, 7645, 6986, 9237, 2129, 2730, 2088, 7167, 5089, 359, 161, 1075, 1673, 9488, 867, 4757, 47, 577, 4505, 2674, 71, 8989, 8064, 408, 4344, 5900, 9958, 8953, 5584, 6920, 1507, 8267, 2199, 2810, 6269, 3955, 8514, 1872, 7414, 1411, 9193, 7794, 4905, 1746, 4491, 533, 9878, 1691, 1232, 3231, 9513, 7337, 1872, 1676, 610, 3889, 8674, 7777, 1552, 3008, 1527, 8927, 3855, 4823, 5176, 2138, 6180, 8500, 5197, 1867, 7094, 5699, 3855, 2114, 544, 612, 7936, 2528, 8612, 2799, 4330, 5332, 3394, 2037, 7430, 852, 1929, 7401, 7694, 6319, 8797, 9776, 6324, 7681, 4330, 9124, 5589, 6518, 1176, 6623, 3576, 3938, 2879, 7430, 3371, 1393, 9981, 9923, 3904, 7978, 4982, 2536, 1844, 3097, 2923, 3569, 9025, 8494, 4286, 6809, 3076, 8544, 533, 1551, 1905, 2821, 4973, 6635, 6933, 262, 3120, 8156, 7885, 8047, 3082, 8164, 5717, 4357, 6297, 5862, 4356, 8123, 691, 8446, 7190, 7862, 7821, 9266, 73, 5312, 3737, 2136, 7404, 6586, 1311, 2248, 3029, 2725, 4383, 2515, 850, 1730, 6821, 708, 7955, 2888, 1502, 6389, 3622, 9319, 7211, 6229, 2386, 2939, 5700, 538, 1304, 547, 9679, 2546, 6822, 754, 4593, 334, 2478, 470, 2019, 784, 56, 4503, 9417, 4388, 2035, 9938, 655, 7965, 3681, 1431, 2964, 290, 724}; int k = 22479; cout << sumOfMaxSubarray(nums, k);</pre>			

Passed all tests! ✓

Chính xác

Diểm cho bài nộp này: 1.00/1.00.

Câu hỏi 5

Chính xác

Điểm 1,00 của 1,00

You are given a list of integers `piles` with `n` elements($1 \leq n \leq 100000$), each element represents the number of notebooks of type `i` (notebook type `i` and notebook type `j` are different from each other if and only if `i ≠ j`). You must put all notebooks to `k` boxes so that the capacity of each box is `r` and each box can contain only one type of notebook.

Request: Implement function:

```
int boxRequire(vector<int>& piles, int k)
```

Where `piles` is the list of notebooks, `k` is the maximum number of boxes you can use. This function returns the minimum capacity of each box `r` so that you can put all notebooks to less than or equal to `k` boxes.

Example:

The list of notebooks is `{6, 4, 3}` and the maximum boxes can be used is 5 (`k = 5`). Therefore, minimum capacity of each box is `3` (using 2 boxes for type `0`, 2 boxes for type `1` and 1 box for type `2`).

Note:

In this exercise, the libraries `iostream`, `string`, `cstring`, `climits`, `utility`, `vector`, `list`, `stack`, `queue`, `map`, `unordered_map`, `set`, `unordered_set`, `functional`, `algorithm` has been included and `namespace std` are used. You can write helper functions and classes. Importing other libraries is allowed, but not encouraged, and may result in unexpected errors.

For example:

Test	Result
<code>int k = 5; vector<int> piles{6,4,3}; cout << boxRequire(piles, k);</code>	3
<code>int k = 11; vector<int> piles{5,1,4,3,5}; cout << boxRequire(piles, k);</code>	2

Answer: (penalty regime: 0, 0, 0, 5, 10, ... %)

Reset answer

```
1 bool check(vector<int>& bananas, int mid_val, int H)
2 {
3     int time = 0;
4     int n = bananas.size();
5     for (int i = 0; i < n; i++) {
6
7         // to get the ceil value
8         if (bananas[i] % mid_val != 0) {
9
10            // in case of odd number
11            time += ((bananas[i] / mid_val) + 1);
12        }
13    else {
14
15        // in case of even number
16        time += (bananas[i] / mid_val);
17    }
18 }
19
20 // check if time is less than
21 // or equals to given hour
22 if (time <= H) {
23     return true;
24 }
25 else {
26     return false;
27 }
28 }
```

```

29
30 int boxRequire(vector<int>& piles, int H)
31 {
32
33     // as minimum speed of eating must be 1
34     int start = 1;
35
36     // Maximum speed of eating
37     // is the maximum bananas in given piles
38     int end = *max_element(piles.begin(), piles.end());
39
40     while (start < end) {
41         int mid = start + (end - start) / 2;
42
43         // Check if the mid(hours) is valid
44         if ((check(piles, mid, H)) == true) {
45
46             // If valid continue to search
47             // lower speed
48             end = mid;
49         }
50     else {
51         // If cant finish bananas in given
52         // hours, then increase the speed
53         start = mid + 1;
54     }
55 }
56
57 }
```

	Test	Expected	Got	
✓	int k = 5; vector<int> piles{6,4,3}; cout << boxRequire(piles, k);	3	3	✓

Passed all tests! ✓

Chỉnh xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 6

Chính xác

Điểm 1,00 của 1,00

You are keeping score for a basketball game with some new rules. The game consists of several rounds, where the scores of past rounds may affect future rounds' scores.

At the beginning of the game, you start with an empty record. You are given a list of strings **ops**, where **ops[i]** is the operation you must apply to the record, with the following rules:

- A non-negative integer **x** (**from 0 to 9**) - record a new score of **x**
- '**+**' - Record a new score that is the sum of the previous two scores. It is guaranteed there will always be two previous scores.
- '**D**' - Record a new score that is double the previous score. It is guaranteed there will always be a previous score.
- '**C**' - Invalidate the previous score, removing it from the record. It is guaranteed there will always be a previous score.

Finally, return the sum of all scores in the record.

For example:

ops = "52CD+"

- '5' - add to the record. Record now is [5]
- '2' - add to the record. Record now is [5,2]
- 'C' - invalid the previous score (2). Record now is [5]
- 'D' - Record new score that is double of previous score (5*2). Record now is [5,10]
- '+' - Record a new score that is the sum of the previous two scores. Record now is [5,10,15]

Return the sum: $5+10+15 = 30$

For example:

Test	Result
cout << baseballScore("52CD+");	30
cout << baseballScore("524CD9++");	55

Answer: (penalty regime: 0, 0, 5, 10 %)

Reset answer

```

1 int baseballScore(string ops){
2 /*TODO*/
3     vector<int> v;
4     int size=ops.size();
5     for(int i=0;i<size;i++){
6         //cout<<ops[i]<<endl;
7         if('0'<=ops[i]&&ops[i]<='9'){
8             v.push_back(ops[i]-48);
9             //cout<<ops[i]<<endl;
10        }
11        else if(ops[i]=='C'){
12            v.pop_back();
13        }
14        else if(ops[i]=='D'){
15            int temp=v.back();
16            //cout<<v.back()<<endl;
17            v.push_back(temp*2);
18        }
19        else if(ops[i]=='+'){
20            int temp=v.back();
21            vector<int>::iterator temp_back=v.end();
22            temp_back--;
23            temp_back--;
24            int temp2=*temp_back;
25            v.push_back(temp+temp2);
}

```

```
26     }
27     //cout<<v[0]<<endl;
28 }
29 int sum=0;
30 v
31 for(unsigned int i=0;i<v.size();i++){
32     sum+= v[i];
33 }
34 return sum;
35 }
```

	Test	Expected	Got	
✓	cout << baseballScore("52CD+");	30	30	✓
✓	cout << baseballScore("524CD9++");	55	55	✓
✓	cout << baseballScore("5C4C2C11+D3");	11	11	✓

Passed all tests! ✓

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 7

Chính xác

Điểm 1,00 của 1,00

Implement all methods in class **Stack** with template type **T**. The description of each method is written as comment in frame code.

```
#ifndef STACK_H
#define STACK_H
#include "DLinkedList.h"
template<class T>
class Stack {
protected:
    DLinkedList<T> list;
public:
    Stack() {}
    void push(T item) ;
    T pop() ;
    T top() ;
    bool empty() ;
    int size() ;
    void clear() ;
};

#endif
```

You can use all methods in class **DLinkedList** without implementing them again. The description of class **DLinkedList** is written as comment in frame code.

```
template <class T>
class DLinkedList
{
public:
    class Node;      //forward declaration
protected:
    Node* head;
    Node* tail;
    int count;
public:
    DLinkedList() ;
    ~DLinkedList();
    void add(const T& e);
    void add(int index, const T& e);
    T removeAt(int index);
    bool removeItem(const T& removeItem);
    bool empty();
    int size();
    void clear();
    T get(int index);
    void set(int index, const T& e);
    int indexOf(const T& item);
    bool contains(const T& item);
};
```

For example:

Test	Result
Stack<int> stack; cout << stack.empty() << " " << stack.size();	1 0
Stack<int> stack; int item[] = { 3, 1, 4, 5, 2, 8, 10, 12 }; for (int idx = 0; idx < 8; idx++) stack.push(item[idx]); assert(stack.top() == 12); stack.pop(); stack.pop(); cout << stack.top();	8

Answer: (penalty regime: 0, 0, 5, 10 %)

[Reset answer](#)

```

1 void push(T item) {
2     // TODO: Push new element into the top of the stack
3     list.add(0,item);
4     //cout<<list.get(0);
5 }
6
7 T pop() {
8     // TODO: Remove an element on top of the stack
9     T ret=list.get(0);
10    list.removeAt(0);
11    return ret;
12 }
13
14 T top() {
15     // TODO: Get value of the element on top of the stack
16     //cout<<list.get(0);
17     return list.get(0);
18 }
19
20 bool empty() {
21     // TODO: Determine if the stack is empty
22     return list.empty();
23 }
24
25 int size() {
26     // TODO: Get the size of the stack
27     return list.size();
28 }
29
30 void clear() {
31     // TODO: Clear all elements of the stack
32     list.clear();
33 }
```



	Test	Expected	Got	
✓	Stack<int> stack; cout << stack.empty() << " " << stack.size();	1 0	1 0	✓
✓	Stack<int> stack; int item[] = { 3, 1, 4, 5, 2, 8, 10, 12 }; for (int idx = 0; idx < 8; idx++) stack.push(item[idx]); assert(stack.top() == 12); stack.pop(); stack.pop(); cout << stack.top();	8	8	✓

Passed all tests! ✓

Chỉnh xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 8

Chính xác

Điểm 1,00 của 1,00

Given an array of positive integers `nums`. An array `mt` represents a mountain having no valleys and taking `nums` as its upperbound. In the other word, for all index `i` in range, `mt[i] ≤ nums[i]` and no pair of indices `(j, k)` that `j < i < k` and `mt[j] > mt[i] && mt[i] < mt[k]` exists.

Your task is to implement a function with following prototype:

```
int mountainWithoutValley(vector<int>& nums);
```

The function returns the maximum sum of numbers of `mt`.

Note:

- The `iostream`, `vector`, `climits` and `stack` libraries have been included and `namespace std` is being used. No other libraries are allowed.
- You can write helper functions.

For example:

Test	Result
<code>vector<int> nums {10, 6, 8, 8}; cout << mountainWithoutValley(nums);</code>	28

Answer: (penalty regime: 0 %)

Reset answer

```

1 int mountainWithoutValley(vector<int>& nums) {
2     // STUDENT ANSWER
3     vector<pair<int,int>> up;
4     vector<pair<int,int>> down;
5     int max=INT_MIN;
6     up.push_back({nums[0],0});
7     //down.push_back({nums[0],0});
8     int size=nums.size();
9     int sum1=nums[0];
10    for(int i=1;i<size;i++){
11        pair<int,int> temp_up_back=up.back();
12        if(nums[i]>=temp_up_back.first){
13            up.push_back({nums[i],i});
14            sum1+=nums[i];
15        }
16        else {
17            int sum2;
18            if(nums[i]*(size-i-1)+sum1<=max){
19                goto skip;
20            }
21            down.push_back({nums[i],i});
22            sum2=nums[i];
23            for(int j=i+1;j<size;j++){
24                pair<int,int> temp_down_back=down.back();
25                if(nums[j]<=temp_down_back.first){
26                    if(sum1+sum2+nums[j]*(size-j-1)<max) {down.clear();goto skip;}
27                    down.push_back({nums[j],j});
28                    sum2+=nums[j];
29                }
30            else{
31                //temp_down_back.second++;
32                down[down.size()-1].second++;
33                sum2+=down[down.size()-1].first;
34            }
35        }
36        if(sum1+sum2>max){
37            max=sum1+sum2;
38        }
39        down.clear();
40    skip:
41    }
42 }
```

```

41 v
42     while(nums[i]<temp_up_back.first){
43         if(up.size()==1){sum1-=up[0].first*(up[0].second+1);}
44         else {
45             int up_size=up.size();
46             sum1 -= up[up_size-1].first*(up[up_size-1].second-up[up_size-2].second);
47         }
48         //sum1-=temp_up_back.first;
49         up.pop_back();
50         if(up.size()==0){
51             break;
52         }
53         temp_up_back=up.back();
54     }
55     up.push_back({nums[i],i});
56     if(up.size()==1)
57         sum1+=nums[i]*(i-0+1);
58     else
59         sum1+=nums[i]*(up[up.size()-1].second-up[up.size()-2].second);
60 }
61 }
62 sum1=up[0].first*(up[0].second+1);
63 int up_size=up.size();//add phần up
64 for(int j=1;j<up_size;j++){
65     sum1=sum1+up[j].first*(up[j].second-up[j-1].second);
66 }
67 //cout<<"sum1_up"<<sum1<<endl;
68 if(sum1>max) max=sum1;
69 return max;
70 }
```

	Test	Expected	Got	
✓	vector<int> nums {10, 6, 8, 8}; cout << mountainWithoutValley(nums);	28	28	✓
✓	vector<int> nums {163}; cout << mountainWithoutValley(nums);	163	163	✓
✓	vector<int> nums {1891, 4518, 1065, 4265, 2554, 2400, 1740, 943, 4162, 2251, 3287, 2706, 2331, 3642, 824, 1233, 795, 1073, 4582, 917}; cout << mountainWithoutValley(nums);	27789	27789	✓
✓	vector<int> nums {4546, 1423, 1446, 4681, 114, 3903, 1973, 3445, 3798, 79, 1726, 4166, 698, 4004, 2143, 3480, 2229, 3501, 751, 2918, 226, 4968, 2666, 2729, 1685, 242, 4417, 4752, 2660, 939, 243, 2527, 4227, 4321, 152, 3915, 1444, 324, 4584, 204, 791, 218, 2567, 4580, 423, 819, 996, 2855, 4885, 1335, 3059, 3356, 225, 4703, 1839, 4131, 1142, 4258, 348, 3663, 2910, 1521, 2924, 1197, 2206, 4869, 4887, 1438, 2706, 3175, 4769, 794, 3244, 264, 983, 3773, 4385, 962, 586, 7, 3100, 4135, 2695, 4510, 4960, 3857, 4030, 1055, 1552, 265, 3618, 4997, 4713, 2579, 3144, 4097, 3563, 2178, 971, 18}; cout << mountainWithoutValley(nums);	37627	37627	✓

Test	Expected	Got	
✓ vector<int> nums {395, 3928, 4696, 3062, 4197, 2593, 4109, 2906, 64, 4113, 3455, 2191, 4994, 1891, 733, 3590, 1086, 4044, 3119, 1126, 378, 2903, 2427, 1275, 3166, 4785, 3240, 3568, 4994, 960, 4692, 144, 2400, 1576, 2188, 2802, 2234, 1878, 1747, 3627, 2404, 2472, 36, 1840, 4765, 4037, 4747, 4900, 3259, 3329, 2569, 1071, 93, 180, 4246, 1576, 4756, 3028, 1205, 4364, 436, 3759, 2146, 1096, 4071, 1021, 2422, 979, 3716, 3925, 735, 4143, 2742, 1920, 3113, 3258, 4391, 3995, 514, 464, 2527, 1225, 4269, 2259, 1480, 168, 2517, 2701, 3436, 1946, 2380, 3109, 4904, 3719, 4324, 3177, 4922, 1385, 158, 4899, 1082, 4608, 4418, 919, 4877, 4783, 3358, 924, 4696, 1327, 3082, 1097, 4099, 1994, 4155, 1233, 867, 1539, 4950, 778, 2316, 4860, 5000, 1603, 4495, 4048, 4796, 484, 1393, 4306, 3044, 4810, 2852, 4861, 2107, 4211, 1234, 374, 3259, 1190, 3742, 3423, 1235, 4175, 646, 3582, 1587, 1410, 662, 3029, 226, 2422, 4206, 1538, 1880, 1284, 3080, 786, 328, 368, 2386, 1802, 4384, 63, 2555, 1758, 4084, 2529, 2235, 2221, 1495, 2818, 3979, 2741, 2492, 2027, 1768, 3147, 2741, 4150, 1050, 1742, 761, 3042, 1610, 3733, 1491, 3235, 75, 3341, 1961, 3438, 431, 3407, 1530, 723, 2029, 1898, 2667, 2644, 4698, 2533, 1332, 1973, 2327, 104, 878, 1309, 3993, 4299, 2216, 4916, 1062, 2078, 475, 4069, 2886, 2411, 1890, 942, 3744, 1185, 1166, 1861, 520, 2339, 257, 3190, 1965, 3017, 1145, 650, 2367, 500, 2848, 1698, 2320, 1399, 1385, 1932, 3010, 397, 1429, 3813, 4738, 1806, 4792, 2960, 2164, 2222, 1253, 4766, 2360, 1623, 1993, 3294, 4571, 1565, 926, 1807, 824, 146, 4310, 2596, 912, 4532, 84, 2975, 3040, 348, 3277, 4833, 4203, 713, 4684, 3888, 2199, 4292, 2871, 2658, 4964, 2564, 644, 3010, 274, 2557, 1152, 1953, 1434, 2197, 4017, 22, 2080, 4455, 2922, 1625, 703, 1716, 1127, 3124, 3574, 1270, 1786, 4203, 4882, 1502, 1495, 62, 1876, 2433, 2609, 2668, 964, 1519, 3479, 3082, 2382, 4725, 164, 309, 1884, 99, 2898, 1996, 1356, 1639, 4245, 3602, 1268, 3891, 4723, 178, 2395, 1317, 1416, 795, 2057, 1006, 3904, 4662, 4978, 262, 2870, 1286, 2310, 3614, 3293, 336, 4661, 3925, 981, 4090, 3626, 3119, 3289, 3225, 981, 3530, 4814, 3182, 4597, 1579, 4198, 3505, 2446, 3241, 4870, 848, 1359, 3376, 1309, 1889, 1023, 1873, 2326, 4527, 306, 896, 3743, 1994, 3094, 4327, 2595, 586, 1291, 17, 1029, 1612, 434, 4256, 2134, 1046, 4076, 4515, 769, 952, 3287, 2090, 1893, 2163, 1421, 2516, 635, 1119, 613, 1964, 2312, 4967, 3182, 1327, 3605, 2152, 2513, 2929, 2198, 2571, 4207, 4305, 2079, 3006, 3667, 3546, 1909, 1665, 2784, 4419, 772, 1167, 22, 1281, 429, 3028, 4497, 3778, 4555, 1367, 4891, 294, 3974, 3440, 3868, 4314, 4007, 2851, 527, 1797, 1789, 3845, 3094, 3175, 2519, 2358, 3480, 3958, 2419, 5000, 730, 1690, 663, 1280, 532, 2060, 4926, 522, 310, 1195, 2557, 4057, 2778, 4991, 1654, 2335, 4451, 1785, 3965, 770, 3591, 3594, 1033, 4769, 1137, 803, 3013, 4797, 450, 1421, 2376, 2035, 4849, 3714, 1481, 288, 1008, 3508, 4103, 2750, 505, 1317, 2017, 2487}; cout << mountainWithoutValley(nums);	70115	70115	✓
✓ vector<int> nums {2041, 1478, 1201, 3999, 1716, 1580, 4846, 4463, 401, 3816, 1634, 3958, 3447, 45, 4881, 4555, 1768, 1927, 1995, 313, 1773, 1555, 4924, 4066, 3722, 305, 4882, 1071, 892, 1609, 430, 126, 456, 1317, 2199, 102, 4373, 3523, 3139, 2655, 3501, 3559, 1264, 3285, 2064, 3510, 4562, 3807, 4098, 3765, 230, 634, 4526, 3795, 38, 3660, 1491, 4428, 3453, 638, 1974, 1520, 2646, 2857, 2416, 595, 1272, 1108, 297, 1118, 453, 3716, 980, 3982, 4635, 3414, 421, 2484, 3082, 2137, 721, 4125, 3071, 4824, 3597, 2163, 992, 4758, 442, 1219, 1977, 2626, 1183, 1944, 1399, 2894, 4551, 4304, 2707, 202, 4821, 389, 3513, 654, 4237, 4079, 4122, 3264, 4949, 241, 2119, 467, 460, 2497, 4711, 193, 3633, 371, 2074, 457, 1711, 1624, 4118, 1468, 112, 763, 4546, 933, 2344, 1428, 2834, 1592, 4675, 505, 1402, 3885, 217, 3978, 1910, 548, 2936, 3731, 2938, 426, 2147, 1199, 309, 1149, 1747, 1883, 2155, 3493, 283, 3881, 44, 388, 48, 2303, 4527, 4007, 1816, 391, 1715, 1581, 1282, 384, 1087, 3463, 4175, 3333, 1151, 4001, 1903, 1892, 1922, 2753, 1231, 921, 2067, 2053, 3673, 3308, 1301, 4009, 2221, 4600, 890, 3633, 1975, 4574, 1669, 2092, 4537, 4146, 3151, 3658, 3530, 395, 1887, 1855, 1911, 2401, 3733, 616, 3529, 106, 1113, 3357, 935, 4646, 3151, 1592, 2613, 1004, 3086, 4022, 3061, 452, 2065, 3707, 1038, 3102, 4190, 4267, 386, 707, 2445, 638, 1164, 123, 1011, 3931, 305, 4806, 3048, 1176, 1422, 3511, 4365, 2793, 2795, 1494, 1002, 1175, 1463, 549, 2815, 723, 492, 3709, 4946, 1908, 1540, 4329, 4888, 4788, 4787, 1995, 3639, 3594, 1927, 3366, 3019, 3668, 4014, 1469, 1223, 4722, 1119, 386, 3423, 4019, 1660, 112, 3492, 1538, 4509, 1197, 4840, 945, 1673, 4475, 2410, 2785, 3017, 4446, 1684, 2553, 2013, 2227, 4507, 4621, 4955, 2606, 2055, 753, 1005, 3908, 1892, 4086, 619, 3763, 2919, 1867, 2384, 2518, 3301, 3904, 737, 1422, 3633, 1159, 2612, 3453, 3308, 1917, 3629, 557, 2618, 1632, 807, 681, 1562, 1840, 4081, 2842, 3455, 1210, 2558, 4244, 1885, 2587, 3778, 890, 2069, 1346, 1442, 1327, 1309, 522, 3827, 4446, 1553, 246, 2629, 3645, 3527, 3670, 1769, 1750, 1386, 201, 2065, 2774, 2820, 4395, 4528, 2217, 1602, 1777, 498, 4662, 611, 642, 2452, 462, 209, 2104, 3079, 1451, 2667, 199, 3849, 2175, 1082, 4494, 1053, 4187, 2154, 668, 4452, 1865, 3499, 3845, 3924, 1219, 2801, 2453, 4520, 3629, 1537, 523, 2728, 2944, 2374, 850, 2658, 2625, 2330, 3428, 2089, 3938, 728, 4464, 573, 192, 2050, 2062, 3827, 1509, 2014, 4328, 3927, 118, 2359, 3323, 599, 4327, 4761, 4974, 3138, 4189, 3420, 699, 4928, 4708, 3517, 140, 4571, 1739, 3009, 4247, 2111, 4365, 884, 4172, 3998, 4369, 1642, 1287, 1904, 2644, 739, 4322, 3833, 3781, 3716, 4037, 4082, 1009, 155, 3640, 4692, 4689, 4227, 727, 1091, 3401, 1890, 1463, 832, 1914, 4175, 1433, 1793, 1674, 1727, 132, 2213, 954, 465, 2376, 2314, 224, 3625, 4113, 2506, 502, 3979, 3256, 214, 2475, 249, 2224, 2657, 137, 2126, 367, 3520, 303, 4075, 1028, 2704, 3175, 3409, 3192, 3096, 410, 1121, 3016, 3537, 1483, 945, 714, 4868, 1945, 3008, 4885, 210, 2681, 1371, 3430, 4842, 3836, 1737, 4457, 2430, 293, 2100, 92, 63, 61, 2625, 3298, 1683, 2530, 650, 679, 4059, 181, 3433, 3447, 3218, 636, 3392, 3556, 451, 1104, 2191, 2688, 1906, 1935, 4774, 1647, 2662, 1964, 1123, 3393, 1341, 3061,	151037	151037	✓

Test	Expected	Got	
<pre>4254, 2246, 1880, 1853, 1596, 2066, 1193, 668, 3185, 4684, 499, 1196, 3464, 3610, 3432, 2845, 472, 3335, 1033, 3827, 983, 751, 3945, 4094, 3186, 1866, 3914, 2635, 345, 2576, 1413, 1633, 1581, 1044, 1212, 3758, 1547, 3844, 1051, 3027, 3421, 640, 2078, 3725, 3055, 1176, 3284, 3990, 4967, 4355, 3053, 4076, 2928, 4405, 143, 903, 2425, 3760, 1615, 3385, 610, 2111, 1063, 2345, 2190, 1415, 2538, 3384, 4008, 1703, 768, 1496, 4423, 3618, 702, 3642, 547, 3146, 3805, 4167, 2549, 3327, 103, 1409, 306, 3928, 3477, 4395, 4497, 3337, 4860, 654, 1861, 4822, 3937, 3517, 3780, 1854, 1955, 1707, 2545, 4204, 2320, 2180, 3839, 1098, 2123, 2417, 255, 1471, 188, 2688, 1244, 4371, 248, 2684, 2437, 1696, 3850, 1976, 2691, 4548, 3290, 946, 2779, 628, 77, 454, 1934, 1570, 1102, 4767, 1159, 1466, 4076, 4862, 1669, 7 ...snip..., 2779, 1928, 4671, 1154, 4876, 2153, 4128, 4511, 4240, 1701, 3631, 4680, 4227, 3083, 3682, 2716, 4573, 4673, 20, 4382, 687, 4251, 3222, 2958, 3377, 3210, 2878, 3206, 3059, 4271, 2008, 4375, 2299, 632, 1317, 444, 2342, 562, 3058, 407, 2570, 4209, 3930, 4740, 1211, 1083, 790, 1835, 804, 2243, 1372, 2785, 1121, 4541, 2550, 2841, 3035, 2493, 2357, 518, 3284, 2549, 1677, 637, 1468, 3518, 4426, 2785, 2354, 1335, 4452, 791, 638, 3524, 2519, 491, 4439, 3784, 3589, 42, 2361, 1599, 3016, 4618, 3441, 1573, 3915, 2055, 612, 2515, 1041, 2103, 4493, 3707, 744, 2888, 2437, 1162, 3231, 816, 3488, 773, 2590, 2709, 494, 535, 3228, 1399, 1479, 2746, 591, 1691, 3842, 4364, 2111, 2129, 4631, 356, 484, 984, 4220, 3293, 4842, 648, 4045, 3858, 1298, 2432, 560, 162, 51, 3625, 3904, 2591, 64, 3811, 151, 1617, 4051, 1143, 2250, 4168, 1386, 1732, 4120, 2650, 514, 1246, 1795, 538, 2042, 1772, 1242, 866, 1601, 3056, 1654, 1522, 4898, 2869, 2155, 913, 206, 1748, 1881, 1835, 547, 4575, 3596, 4707, 4908, 4850, 1257, 3891, 3880, 2951, 267, 4595, 936, 2649, 4388, 3638, 1147, 467, 567, 4531, 1621, 3217, 4918, 2626, 256, 3399, 3351, 4293, 3176, 473, 3053, 4801, 486, 3315, 1852, 3229, 949, 1866, 892, 163, 3217, 3194, 4162, 2124, 1857, 698, 1988, 4121, 370, 3181, 2587, 4607, 1062, 3698, 3790, 507, 556, 1054, 3476, 510, 2602, 1000, 3706, 3586, 4641, 1174, 1910, 4398, 3184, 750, 4203, 1407, 1691, 3679, 3119, 2656, 381, 3969, 4289, 3322, 4428, 3322, 358, 207, 1852, 3515, 3341, 372, 2831, 698, 594, 262, 1819, 4281, 1945, 3766, 1084, 2127, 3964, 4223, 84, 2046, 1054, 2383, 3304, 2917, 952, 2933, 1077, 254, 3875, 2218, 4747, 3202, 2280, 3699, 4008, 4204, 997, 4838, 4656, 2796, 4809, 2498, 1892, 2717, 4298, 20, 3432, 1288, 780, 4199, 2586, 4442, 4196, 3564, 4207, 3324, 758, 2419, 2136, 1562, 2916, 4729, 2134, 2430, 4952, 1242, 3313, 3213, 4836, 1696, 1919, 2874, 1656, 500, 2611, 154, 1876, 1669, 1380, 2760, 3327, 4615, 249, 1914, 967, 4526, 25, 4076, 3801, 4371, 1983, 1347, 1467, 1110, 4637, 676, 3176, 2253, 1408, 2206, 2524, 2843, 1053, 4829, 944, 2500, 3267, 4657, 913, 3208, 4633, 2319, 3540, 4015, 97, 4350, 4893, 2252, 2997, 485, 664, 837, 968, 2282, 4703, 1793, 1921, 1518, 1558, 2171, 2932, 2063, 1843, 350, 1801, 4854, 1217, 3826, 4105, 3507, 4991, 3774, 1742, 269, 2629, 2502, 546, 854, 783, 1671, 371, 2927, 637, 2120, 837, 1256, 153, 2388, 889, 290, 674, 2461, 613, 2880, 697, 931, 2219, 275, 291, 293, 335, 3198, 3351, 2143, 1803, 818, 536, 3436, 245, 2515, 4636, 1278, 475, 4610, 1750, 4549, 3566, 1603, 2697, 2604, 2829, 3466, 2070, 3378, 715, 1202, 4880, 935, 3140, 2950, 1414, 973, 2396, 1676, 4817, 4498, 1688, 2166, 3710, 667, 1465, 1867, 1502, 282, 4950, 4342, 3540, 475, 4753, 3645, 639, 2332, 3502, 348, 1561, 3435, 2591, 3323, 4766, 4660, 1475, 870, 4873, 1499, 2120, 3867, 1100, 4431, 4886, 1417, 4567, 136, 4478, 127, 2619, 1995, 3211, 1531, 1511, 3275, 232, 1381, 1866, 3329, 283, 2538, 1466, 2167, 3592, 1453, 1389, 4970, 4680, 4323, 4694, 759, 1035, 613, 1266, 2407, 4356, 3508, 2027, 4974, 413, 3800, 3276, 440, 3523, 2881, 439, 2348, 4458, 3562, 3852, 302, 556, 539, 2306, 136, 4694, 2014, 2595, 3208, 1292, 1964, 3635, 896, 2598, 273, 4497, 3530, 211, 3437, 1513, 3170, 2861, 4764, 2185, 977, 1389, 3465, 3835, 887, 2120, 2622, 4723, 4825, 3321, 1615, 1105, 3923, 4338, 1184, 1889, 2055, 4854, 283, 425, 3383, 1657, 2655, 98, 3615, 4265, 1805, 4419, 2671, 110, 2421, 4874, 1375, 1697, 3417, 1180, 4804, 2065, 2440, 496, 346, 2051, 1934, 1927, 3906, 123, 1049, 2894, 2135, 2758, 831, 831, 1875, 1058, 711, 2493, 4815, 2720, 1509, 1407, 2979, 1841, 2082, 4759, 4578, 3807, 3474, 3459, 4338, 69, 3880, 4003, 3258, 4602, 4960, 1500, 949, 4345, 683, 1588, 3175, 2404, 2992, 3823, 2661, 2275, 3216, 3994, 4458, 1784, 2115, 1342, 4622, 197, 1578, 911, 401, 1464, 2205, 926, 754, 1069, 4790, 963, 4057, 3836, 1989, 1359, 4707, 1229, 3373, 1297, 615, 2809, 2286, 2639, 4212, 4243, 1556, 960, 1742, 975, 2185, 128, 1385, 1482, 4655, 1451, 3837}; cout << mountainWithoutValley(nums);</pre>			
✓	vector<int> nums {651, 3777, 2037, 267, 2463, 3930, 2218, 3950, 931, 2685, 1241, 3629, 1926, 1086, 1006, 3464, 764, 1529, 1488, 1561, 2291, 4635, 944, 3814, 3663, 4181, 4636, 844, 3929, 2040, 155, 3982, 1200, 2421, 4522, 1726, 814, 4812, 1494, 482, 374, 4008, 154, 1651, 1748, 3312, 1515, 1442, 838, 3601, 3850, 764, 4209, 2803, 3851, 4688, 456, 633, 476, 3830, 464, 909, 919, 643, 719, 203, 4307, 1162, 4607, 1860, 4336, 3605, 2197, 1286, 842, 4268, 3471, 450, 2048, 1140, 3678, 287, 1405, 3900, 1952, 629, 1399, 253, 808, 1364, 1990, 2200, 3178, 422, 2759, 2732, 2897, 2554, 2129, 1168, 4796, 2018, 4712, 1213, 2513, 4911, 4249, 4696, 1264, 2116, 1413, 4327, 3244, 129, 105, 4074, 2165, 1361, 1500, 4340, 1953, 102, 563, 1601, 3852, 3419, 4274, 1034, 4198, 4969, 4156, 359, 3188, 910, 3405, 2541, 2977, 3644, 1126, 3248, 1758, 4910, 3537, 4107, 1808, 576, 2405, 2645, 658, 231, 1116, 422, 4970, 2871, 3011, 938, 3432, 1956, 4722, 4813, 2164, 3010, 3596, 608, 2359, 609, 1857, 2531, 3022, 1307, 3194, 188, 746, 4112, 237, 4294, 4419, 3605, 3962, 4766, 702, 2224, 2861, 1074, 3180, 3527, 3190, 2907, 325, 2841, 709, 4668, 4885, 1140, 4089, 347, 3510, 169, 3858, 2464, 3158, 1790, 3141, 107, 3715, 1986, 145, 2811, 4436, 2326, 2472, 1612, 4193, 2698, 2240, 2353, 4286, 4294, 3155, 691, 2100, 1026, 1196, 840, 3760, 2815, 4082, 727, 4768, 96, 3321, 3795, 2211, 2347, 159, 854, 2545, 2245, 317, 520, 1299, 2149, 3831, 2616, 3260, 4535, 3806, 3882, 464, 4692, 4235,	161547	161547 ✓

Test	Expected	Got
1802, 483, 3983, 3030, 4280, 734, 3163, 2993, 4997, 1725, 2876, 4278, 3157, 1780, 3068, 4944, 764, 4348, 583, 2393, 158, 4457, 4080, 439, 1002, 1422, 4293, 3300, 462, 506, 2609, 1494, 3705, 1900, 633, 3181, 2277, 1061, 4609, 1204, 3588, 1366, 4679, 345, 4053, 2314, 1871, 4696, 2502, 2629, 1377, 3132, 1332, 2205, 4729, 1735, 3249, 2431, 1054, 3285, 2069, 981, 997, 964, 960, 2471, 806, 967, 2295, 2775, 2407, 604, 4029, 3138, 887, 3561, 1860, 2036, 4510, 4814, 1988, 2819, 988, 4951, 4363, 3383, 3535, 2057, 2397, 896, 1505, 1975, 550, 1768, 3408, 3485, 3614, 242, 4623, 829, 4205, 524, 2414, 2352, 4335, 649, 1102, 552, 4897, 4714, 4568, 2739, 3397, 582, 2222, 2082, 906, 143, 2234, 1083, 4208, 1656, 3157, 4404, 162, 1059, 1227, 4793, 4972, 1887, 349, 2327, 1983, 3963, 550, 810, 3547, 4436, 4675, 292, 3357, 2429, 1229, 4111, 1929, 2439, 2105, 3877, 562, 4646, 1465, 1158, 999, 3711, 3900, 1258, 2032, 4823, 1436, 967, 4898, 4846, 3767, 141, 318, 1060, 1104, 2376, 2265, 1962, 1334, 2414, 2608, 1108, 3741, 3842, 4346, 141, 4107, 344, 2034, 2899, 1398, 4434, 1117, 3525, 1373, 3346, 2465, 643, 3316, 2984, 3743, 2413, 3804, 1303, 3002, 327, 1194, 3883, 3808, 3155, 1095, 3902, 2121, 3919, 4720, 2811, 2893, 1832, 3686, 3842, 1971, 1991, 2800, 726, 3392, 3419, 2432, 3827, 172, 576, 4516, 4396, 1187, 350, 4678, 3500, 3466, 4125, 2470, 4203, 3057, 1567, 4886, 4091, 4312, 110, 1846, 1172, 845, 889, 2745, 1317, 1661, 3481, 4807, 2427, 984, 1480, 4845, 983, 738, 1060, 514, 4187, 1863, 1344, 980, 1237, 3087, 4821, 1018, 944, 915, 3215, 3387, 207, 703, 2242, 631, 2275, 2496, 4311, 2404, 3135, 1495, 3313, 3852, 2362, 4774, 4421, 3900, 3212, 2155, 777, 4624, 613, 4731, 2244, 2589, 2442, 113, 863, 4152, 2681, 3775, 1042, 3190, 4056, 755, 3954, 2187, 4281, 3702, 3794, 2998, 1481, 4810, 207, 4733, 4857, 2202, 881, 797, 1267, 4822, 22, 757, 665, 2006, 129, 299, 4091, 2431, 1243, 3623, 1728, 4588, 4921, 639, 871, 807, 3359, 458, 3014, 808, 3498, 3545, 1426, 4581, 3428, 4547, 3512, 3963, 4196, 4256, 1556, 1011, 3564, 120, 72, 4161, 2935, 1885, 1584, 337, 3266, 174, 203, 4563, 3883, 1294, 2700, 1833, 719, 4453, 4106, 1182, 3976, 44, 3164, 1193, 2783, 3758, 2897, 4545, 2398, 3909, 2303, 940, 3322, 59, 1066, 3227, 3831, 1571, 4904, 1536, 4780, 3455, 659, 3699, 2192, 3859, 4479, 2753, 1308, 4314, 1388, 3215, 2707, 479, 2005, 4133, 4697, 3431, 1314, 3926, 4912, 3605, 2388, 1158, 688, 213, 1847, 1725, 143, 2797, 4693, 604, 1943, 409, 4384, 2485, 4968, 2327, 1263, 1056, 3174, 2482, 197, 4091, 4636, 1514, 615, 3602, 64, 4156, ...snip... 63, 2392, 4764, 969, 444, 416, 483, 3962, 3977, 1257, 1807, 39, 2068, 3242, 2476, 3744, 1171, 3219, 3546, 3849, 1150, 1063, 616, 3465, 1568, 1683, 704, 2959, 2374, 4118, 251, 2818, 1740, 609, 894, 503, 2340, 145, 557, 4953, 3048, 3467, 71, 1683, 1074, 2615, 2610, 3338, 4466, 4048, 2313, 1825, 2952, 827, 4635, 4316, 434, 2542, 3302, 2370, 4168, 4099, 3891, 2693, 1342, 2308, 3300, 4891, 154, 4521, 3419, 4714, 3414, 216, 4981, 1439, 1240, 1573, 3191, 4080, 4081, 2235, 3975, 2283, 4427, 3065, 3439, 4814, 4518, 2496, 2749, 4449, 1701, 3484, 70, 4841, 3212, 3153, 2645, 422, 445, 3128, 2350, 438, 1067, 26, 2560, 4339, 4502, 3572, 2578, 3547, 2713, 3419, 3197, 3888, 544, 530, 1962, 4415, 3646, 266, 1614, 4684, 2238, 282, 4651, 2718, 3032, 4831, 361, 2598, 1590, 4653, 3405, 4382, 4663, 3510, 1920, 1864, 1844, 1707, 2443, 452, 1258, 2569, 4873, 4518, 1927, 1149, 151, 2639, 4144, 2828, 2159, 2779, 2090, 2140, 115, 1452, 2448, 2287, 4272, 1141, 2621, 2961, 3184, 3329, 2666, 4543, 2941, 1404, 3210, 3714, 3, 3034, 2651, 2909, 2525, 735, 3248, 4723, 4027, 142, 1077, 3181, 3042, 2655, 3227, 4651, 3949, 3467, 4613, 1488, 3594, 2753, 458, 1555, 2136, 3502, 1739, 251, 1360, 640, 2591, 4738, 1321, 132, 262, 3150, 1601, 1081, 2603, 3460, 2928, 2109, 3245, 3117, 383, 1910, 2735, 2510, 1953, 1787, 1415, 2731, 544, 52, 3836, 1348, 716, 119, 588, 1554, 3144, 2629, 2404, 1795, 4701, 3420, 4497, 3564, 11, 550, 439, 4207, 1733, 4858, 952, 4918, 1692, 129, 2287, 2822, 1007, 1732, 1246, 3495, 962, 2637, 2148, 123, 3228, 4531, 1115, 397, 2885, 3481, 1345, 2076, 2296, 705, 1733, 448, 2597, 1117, 4557, 3453, 866, 2006, 470, 2422, 4715, 1113, 1443, 99, 768, 1991, 4001, 589, 3150, 2091, 4907, 841, 1753, 922, 3961, 1980, 3176, 4686, 4838, 4646, 1801, 1555, 2618, 1122, 590, 670, 3413, 2861, 1832, 2770, 2342, 2725, 3588, 108, 2651, 1260, 1971, 4618, 545, 4613, 2342, 3680, 3439, 1102, 1681, 3539, 1806, 294, 232, 2482, 2355, 1513, 1456, 1674, 3232, 3394, 2687, 4616, 4511, 3736, 1142, 1964, 313, 1180, 2141, 4348, 175, 2330, 360, 1198, 479, 1118, 161, 1883, 4728, 3145, 3951, 5000, 2761, 2588, 4965, 2327, 719, 3788, 2813, 1830, 4462, 4398, 4511, 4277, 1547, 4253, 4692, 2238, 2935, 2466, 600, 1976, 4470, 2276, 3296, 3916, 4905, 520, 696, 3164, 3648, 1618, 2808, 216, 4748, 4901, 216, 1124, 180, 2759, 3505, 1159, 398, 1225, 846, 724, 2801, 533, 2851, 1302, 1960, 1217, 4420, 3656, 2273, 1939, 914, 3323, 4199, 421, 4962, 2667, 4653, 811, 3582, 2468, 1117, 1530, 1288, 2224, 568, 1587, 1171, 3325, 1630, 825, 1172, 3709, 3904, 32, 4834, 238, 108, 1390, 3218, 3731, 3744, 2334, 1126, 34, 1939, 514, 3135, 2520, 2321, 185, 1608, 634, 4493, 562, 4341, 3733, 4604, 840, 4532, 3128, 3023, 102, 4130, 2200, 3377, 3685, 3169, 2420, 2138, 203, 769, 2995, 4163, 1350, 1229, 4731, 435, 986, 623, 3531, 1346, 1079, 3838, 355, 4004, 3342, 3535, 1898, 1824, 1261, 618, 432, 4873, 4754, 823, 2150, 2628, 996, 393, 2362, 3779, 3128, 1081, 4223, 3394, 1515, 222, 1461, 4040, 275, 923, 4029, 4313, 1699, 1697, 1711, 1977, 6, 4423, 3255, 187, 467, 2007, 701, 4515, 4266, 1516, 3189, 2113, 2183, 3425, 4103, 2263, 2196, 4019, 1639, 4188, 1068, 3721, 2103, 3074, 1149, 3298, 2716, 621, 965, 992, 3775, 550, 3945, 2252, 252, 7, 2946, 1819, 4464, 3023, 1385, 631, 1934, 3479, 1563, 889, 2819, 3558, 4928, 4677, 1451, 1651, 1817, 654, 1371, 3825, 4097, 3405, 1353, 864, 1902, 2540, 2289, 1753, 4576, 4199, 2321, 1714, 3897, 4253, 615, 2435, 4876, 2826, 1450, 192, 867, 3561, 3750, 2368, 404, 2314, 1315, 3773, 748, 494, 4040, 1853, 3399, 1058, 2077, 1912, 2710, 467, 1990, 731, 2338, 4367, 2575, 2557, 392, 4053, 3116, 1296, 2302, 3613, 4015, 1370, 1997, 1690, 4968, 4900, 423, 3647, 2568, 3667, 2786, 2720, 3254, 1453, 4894, 1307, 2310, 4598, 692, 2215, 4287,		

Test	Expected	Got	
4756, 1464, 1020, 1108, 3744, 3608, 3739, 3757, 2422, 3941, 1502, 3359, 718, 3729, 2139, 145, 3332, 1113, 1254, 1296, 4028, 1430, 3429, 4989, 3427, 4375, 2824, 4854, 1034, 1097, 1304, 115, 908, 2209, 4886, 2233, 2762, 433}; cout << mountainWithoutValley(nums);			
✓ vector<int> nums {2447, 2164, 4061, 4937, 3631, 192, 2596, 2248, 2317, 785, 4110, 4092, 3478, 2268, 3210, 886, 1492, 286, 1773, 1961, 4511, 27, 4524, 3812, 3492, 3777, 2, 2550, 3086, 1710, 525, 4626, 4322, 1328, 1500, 4851, 136, 1676, 411, 610, 4639, 1224, 187, 2533, 4519, 5, 4147, 2234, 1861, 783, 2532, 1029, 3898, 2367, 2419, 598, 1157, 2411, 1592, 2558, 2971, 2284, 295, 4588, 1820, 1536, 3764, 875, 4239, 1953, 1533, 4814, 3137, 862, 4946, 2335, 3810, 473, 4896, 1731, 4496, 953, 4943, 978, 976, 88, 497, 3390, 4450, 3216, 3964, 2795, 517, 2609, 4921, 1126, 2939, 4214, 3613, 2384, 2166, 3313, 2376, 2733, 4619, 236, 3813, 951, 2174, 554, 1937, 1969, 3206, 868, 288, 2233, 434, 624, 1884, 3765, 1836, 1760, 4143, 2074, 1397, 3863, 675, 225, 4529, 2545, 2543, 315, 4300, 3331, 2245, 1878, 3670, 2176, 946, 4632, 131, 1878, 262, 1251, 2342, 665, 3045, 1242, 583, 3390, 4293, 2939, 2322, 906, 657, 773, 1514, 3132, 3412, 1436, 677, 1385, 786, 4395, 2954, 495, 2796, 2706, 4257, 116, 4243, 2344, 2898, 3063, 1591, 1986, 4032, 3846, 1967, 476, 2735, 1482, 1092, 2303, 996, 4656, 3537, 2524, 4860, 2258, 4110, 560, 4856, 118, 78, 2609, 4599, 3045, 1341, 4240, 4820, 318, 334, 4615, 4850, 1403, 1691, 2293, 4253, 3445, 4542, 664, 1827, 4980, 3000, 2354, 722, 1537, 3785, 4214, 660, 1760, 3863, 4733, 2777, 3728, 3658, 3428, 3171, 2901, 2543, 3985, 4880, 1880, 123, 230, 4523, 644, 4219, 1093, 4524, 1396, 1458, 4093, 2489, 412, 859, 1531, 2677, 3982, 1135, 4292, 601, 1374, 1120, 730, 1232, 31, 3503, 2421, 1767, 3431, 652, 3549, 2804, 2264, 2716, 1765, 2917, 1513, 2951, 3553, 1373, 1933, 3210, 1308, 1284, 3522, 3395, 2379, 4116, 1181, 132, 4462, 4459, 51, 4985, 1040, 4789, 211, 2557, 2876, 409, 2755, 2802, 1947, 1968, 3464, 3719, 2819, 2961, 1186, 4015, 1178, 713, 4887, 1264, 3340, 2233, 4270, 1153, 385, 2304, 1231, 216, 2451, 2778, 961, 4146, 4486, 4633, 4980, 268, 1872, 3819, 4905, 1786, 603, 1293, 839, 3014, 249, 1743, 3823, 3659, 648, 695, 153, 282, 3318, 218, 4064, 333, 142, 3352, 4524, 2866, 1475, 2495, 1487, 4643, 69, 1955, 1187, 355, 4746, 3645, 4201, 4443, 2814, 3545, 3117, 3679, 1934, 1045, 32, 3786, 514, 551, 997, 4737, 4321, 886, 3529, 4487, 4185, 1148, 2053, 3836, 4615, 4705, 461, 939, 1428, 1259, 3510, 4740, 625, 2926, 2572, 1039, 2052, 4271, 3553, 835, 1779, 4581, 4101, 350, 2272, 2825, 746, 2301, 3758, 741, 271, 54, 4156, 660, 2640, 2076, 4367, 2969, 932, 2083, 1844, 2594, 3927, 4613, 4662, 2166, 5000, 1420, 1429, 92, 1541, 33, 2480, 222, 2838, 2725, 3756, 2838, 560, 4918, 4525, 3752, 2811, 827, 4636, 2401, 2942, 3043, 3090, 2774, 4626, 4475, 2541, 4855, 548, 2135, 3032, 1565, 4614, 3324, 3695, 2642, 1000, 1635, 4988, 536, 3963, 408, 739, 4757, 4967, 2731, 3309, 814, 4422, 2950, 4315, 4658, 216, 3161, 3097, 1000, 792, 2403, 1078, 4178, 3783, 3738, 168, 2716, 4295, 1837, 4013, 168, 1986, 3663, 4047, 2788, 2181, 1942, 1408, 527, 791, 157, 55, 4751, 2590, 3223, 2057, 2335, 1781, 3662, 3259, 4759, 2358, 2507, 2959, 3316, 4638, 4054, 4294, 1906, 3369, 2167, 4949, 3159, 4620, 4486, 4890, 633, 3642, 4833, 3049, 4867, 3224, 3116, 3498, 316, 717, 3030, 1365, 4808, 556, 3441, 1241, 2556, 1469, 4314, 561, 2796, 4723, 4754, 1889, 2351, 2614, 4644, 3175, 3717, 4620, 4600, 4309, 192, 1483, 2863, 894, 1881, 2523, 3496, 3399, 1956, 1954, 1107, 384, 3606, 2295, 3436, 195, 4822, 3947, 2473, 3668, 3690, 709, 3479, 3225, 4196, 2021, 1947, 1077, 3957, 2771, 4845, 2351, 3975, 2521, 1009, 1512, 145, 4542, 1921, 5, 2743, 3668, 2381, 2434, 4729, 1117, 3317, 4296, 4453, 3137, 3448, 4471, 3151, 2635, 1606, 1832, 2279, 798, 179, 140, 2068, 3225, 1492, 2321, 4728, 554, 3831, 4951, 2620, 202, 518, 2893, 3750, 1909, 3061, 2706, 3023, 2902, 1493, 1513, 3506, 528, 4895, 2035, 1106, 71, 4244, 27, 2835, 136, 1725, 2031, 1788, 199, 2319, 1851, 1587, 832, 473, 2594, 3969, 1560, 689, 2127, 2440, 1940, 11, 1032, 3882, 202, 2875, 837, 3196, 2895, 1286, 600, 1331, 1488, 780, 1157, 2325, 4909, 161, 2515, 2396, 739, 1604, 1292, 614, 585, 1271, 3770, 1008, 2430, 2032, 1562, ...snip... 8, 2177, 3255, 3503, 3213, 254, 1622, 1620, 261, 1986, 1376, 3985, 1258, 4840, 1113, 271, 386, 2399, 2963, 2340, 286, 687, 3590, 4275, 4125, 3109, 4648, 1083, 509, 1675, 3707, 1797, 1251, 2091, 4991, 202, 2646, 189, 936, 4252, 4183, 1567, 4001, 4204, 1355, 1653, 720, 2852, 3884, 201, 1452, 2621, 3076, 1628, 4413, 940, 1340, 2166, 4066, 4089, 1581, 3627, 4241, 2032, 2725, 391, 947, 3971, 3468, 1547, 4205, 880, 1831, 2119, 3327, 4399, 748, 2370, 2078, 863, 127, 2585, 4356, 4553, 1765, 4564, 2419, 1548, 2383, 991, 1510, 1659, 897, 664, 697, 1508, 2683, 148, 1170, 4467, 3208, 1631, 1639, 4070, 1014, 2875, 386, 2434, 2341, 3704, 3243, 3326, 1395, 366, 4652, 4630, 2823, 3271, 1171, 4235, 4428, 176, 627, 2113, 4457, 1817, 1404, 1201, 4159, 4923, 2100, 24, 2282, 3457, 3990, 832, 1843, 1322, 3138, 4853, 1389, 4239, 45, 2635, 1472, 2616, 2243, 3566, 75, 1700, 3610, 505, 1156, 4640, 3689, 755, 76, 3527, 3000, 2152, 3919, 2067, 4338, 428, 1226, 4147, 2016, 4933, 2736, 3205, 1472, 4649, 1574, 3270, 2166, 4013, 2877, 2200, 4497, 2983, 232, 2907, 4922, 231, 519, 4256, 3200, 3802, 692, 1873, 2135, 999, 3004, 2725, 1483, 2982, 2391, 459, 863, 3508, 643, 1364, 2203, 2332, 622, 4198, 4293, 3283, 2403, 3542, 2407, 4795, 2360, 2893, 834, 43, 3669, 455, 3057, 4988, 2663, 344, 2073, 4983, 3421, 1296, 2991, 3133, 1555, 2097, 147, 3759, 285, 3054, 4611, 4328, 1918, 2830, 2195, 3567, 2274, 1017, 2196, 4057, 3409, 4530, 1458, 2710, 1030, 694, 1851, 3963, 3630, 2068, 3853, 3640, 916, 437, 782, 1083, 1308, 2870, 3866, 1253, 4222, 396, 3833, 1275, 3966, 4615, 4951, 1352, 4522, 3330, 1480, 2809, 4916, 2561, 3611, 3833, 1761, 684, 2870, 125, 2424, 2580, 1827, 2643, 2735, 1596, 3705, 1247, 435, 3994, 3262, 4390, 3214, 997, 3004, 799, 2622, 2408, 2185, 3190, 1611, 2440, 2571, 2843, 1305, 211, 724, 1842, 991, 4930, 795, 3788, 2463, 4289, 3123, 4958, 4863, 2780, 303, 1721, 1767, 1860, 3422, 1098, 2293,	177420	177420	✓

Test	Expected	Got
<pre> 721, 1652, 4202, 1629, 3706, 1298, 1769, 3045, 3564, 4710, 4094, 4888, 1504, 4541, 3003, 352, 1192, 1, 15, 4513, 2546, 3718, 4155, 4788, 4181, 513, 4817, 3258, 4732, 4979, 3745, 2458, 1994, 1770, 2006, 1549, 1181, 4072, 3769, 2063, 709, 1832, 1252, 4792, 3943, 1815, 1410, 3516, 3789, 4560, 2549, 5000, 4576, 2363, 1632, 1502, 857, 3966, 2063, 1283, 516, 1339, 923, 1214, 494, 3972, 2250, 1364, 1608, 1765, 310, 2338, 1676, 2034, 2685, 802, 1368, 3145, 1761, 1819, 1993, 2704, 2656, 4930, 2242, 3890, 2159, 2197, 581, 2222, 4739, 1265, 57, 1993, 2896, 4801, 157, 4109, 582, 4643, 4708, 1713, 2040, 2886, 4753, 2571, 4279, 55, 564, 1907, 3814, 3636, 4942, 4594, 1975, 3200, 4817, 3139, 31, 568, 4924, 4305, 2269, 2053, 542, 2506, 4456, 963, 1480, 2186, 2938, 609, 2072, 2331, 2439, 2619, 2019, 931, 67, 4805, 2955, 2558, 699, 3848, 1721, 85, 2434, 1057, 1213, 1567, 679, 2352, 3708, 2864, 3829, 2079, 2097, 688, 1532, 1180, 447, 2373, 1702, 304, 4933, 2579, 3356, 257, 3811, 994, 1065, 196, 1836, 97, 607, 2767, 213, 3730, 4105, 2639, 517, 638, 4601, 3717, 3317, 3010, 43, 2662, 4490, 962, 2346, 667, 1833, 4069, 706, 2654, 1930, 1425, 3933, 2719, 2024, 801, 273, 958, 2912, 2300, 4562, 513, 2088, 936, 1812, 4432, 1458, 126, 124, 103, 2856, 1784, 752, 1940, 2130, 528, 3491, 1146, 1438, 299, 230, 4477, 1880, 2222, 4452, 3966, 4043, 1391, 2140, 1937, 2062, 791, 1420, 3174, 2274, 2879, 2363, 277, 1867, 889, 1176, 3454, 3525, 2616, 2242, 2007, 4065, 2131, 3856, 3071, 1088, 1549, 2538, 2171, 471, 3329, 2487, 3390, 3275, 2428, 4792, 870, 2039, 2739, 1173, 2805, 4492, 4359, 711, 4670, 2521, 4326, 1749, 1145, 1475, 1940, 2787, 1240, 1750, 2362, 2927, 635, 3628, 4702, 4110, 2792, 262, 3650, 3448, 2423, 4689, 3273, 1111, 613, 3002, 855, 573, 736, 2950, 4873, 1558, 1691, 1603, 4614, 1435, 3129, 4419, 2580, 2105, 3671, 1818, 2821, 1330, 4533, 165, 4635, 2874, 214, 4525, 3170, 1109, 3465, 3741, 2445, 2655, 3606, 4752, 3396, 2508, 2029, 1183, 844, 1593, 539, 1402, 2007, 426, 4790, 3324, 3750, 1366, 3452, 4, 3421, 1448, 1431, 593, 2086, 3207, 3963, 2869}; cout << mountainWithoutValley(nums); </pre>		
✓ <pre> vector<int> nums {2416, 2080, 419, 1910, 928, 738, 2945, 4497, 2465, 2698, 485, 2680, 3393, 649, 3304, 4381, 76, 1747, 3016, 719, 4900, 3720, 2086, 2221, 2921, 1800, 31, 832, 254, 1467, 3689, 2846, 228, 264, 3158, 4468, 3867, 351, 1309, 4084, 2406, 1927, 1407, 3024, 2798, 1076, 2192, 413, 698, 716, 3967, 1851, 2406, 3769, 3967, 1252, 2321, 3204, 2238, 59, 4267, 464, 3622, 3880, 991, 3468, 3595, 2713, 565, 2933, 4268, 2, 4179, 4348, 1069, 520, 1269, 3907, 1228, 3474, 2995, 3790, 2002, 1297, 4034, 328, 3069, 2826, 4237, 379, 3692, 1242, 1182, 2220, 3651, 2801, 167, 3494, 1839, 1194, 3326, 4678, 2508, 1682, 1521, 1407, 13, 31, 710, 2159, 2859, 1019, 470, 4722, 755, 1353, 18, 1429, 4025, 1775, 3526, 2037, 2073, 3372, 4982, 2977, 479, 1084, 3231, 1581, 2277, 490, 2543, 457, 1302, 2661, 1479, 154, 4752, 1910, 4415, 295, 1887, 1305, 4403, 3398, 4824, 874, 2517, 1552, 1051, 338, 4427, 3562, 3261, 3698, 1579, 422, 2004, 375, 4500, 729, 2492, 4966, 1607, 4358, 4345, 3882, 3877, 3771, 3302, 3069, 3682, 4226, 462, 3299, 1619, 2004, 3082, 1973, 2852, 1625, 1250, 932, 653, 2634, 4639, 491, 3814, 1458, 168, 4194, 1072, 2306, 4339, 2249, 2534, 1917, 2151, 930, 156, 4436, 183, 3675, 662, 1850, 3626, 610, 4485, 554, 3137, 553, 4270, 2328, 4080, 76, 2531, 4765, 96, 816, 4221, 2207, 4804, 703, 1122, 2744, 1179, 3753, 2572, 1190, 350, 1797, 3995, 995, 883, 2152, 2740, 2510, 1001, 1485, 4217, 4666, 796, 775, 2483, 3021, 1883, 4467, 3165, 3299, 4488, 2760, 4499, 236, 3632, 3235, 3370, 4611, 4572, 4150, 3130, 609, 2999, 1882, 2922, 4594, 2393, 717, 1459, 4766, 2868, 1045, 4808, 554, 1677, 590, 2363, 48, 4356, 976, 3971, 2716, 4175, 2668, 3269, 2865, 1724, 2000, 4313, 2236, 886, 1370, 2543, 1126, 3202, 3994, 2242, 1440, 443, 378, 2523, 1394, 1858, 2890, 792, 2268, 1826, 1207, 175, 3433, 3547, 44, 1910, 1489, 4789, 2908, 3853, 558, 1503, 150, 2062, 1617, 1921, 434, 539, 141, 4719, 44, 3337, 1645, 2263, 2631, 1332, 1618, 725, 1419, 2788, 1151, 1564, 2089, 182, 184, 4928, 3358, 2789, 2528, 2393, 1871, 4266, 1299, 1051, 2173, 4738, 515, 980, 3553, 4847, 1503, 3662, 2121, 370, 2080, 1053, 2762, 751, 1110, 4571, 3977, 1953, 1915, 483, 4261, 3040, 3758, 250, 1175, 1888, 507, 92, 4510, 3381, 3984, 262, 2732, 3771, 59, 2872, 4492, 4586, 3274, 1262, 2488, 1346, 755, 2055, 4318, 2267, 294, 3621, 3083, 4993, 4872, 838, 4771, 3566, 1825, 226, 3927, 1640, 1232, 1959, 3496, 3604, 29, 2457, 1667, 3907, 1975, 1677, 3582, 805, 504, 3872, 1978, 2051, 3509, 3708, 169, 2065, 1042, 987, 2594, 1923, 398, 2463, 1154, 4039, 4043, 4616, 4453, 3948, 2809, 3567, 294, 2665, 3994, 4, 2923, 434, 3057, 4139, 3869, 1478, 1919, 2964, 2670, 1044, 3478, 955, 3140, 3792, 1138, 2133, 2077, 3840, 1701, 2852, 2507, 3667, 978, 3836, 3002, 4564, 1708, 1999, 159, 4060, 2960, 4230, 620, 4718, 695, 1810, 2193, 4279, 3233, 130, 2981, 3502, 4935, 2477, 2614, 4720, 3655, 410, 205, 1846, 357, 2078, 3413, 3051, 2952, 2886, 3925, 2553, 1699, 302, 4372, 2676, 4917, 4977, 3517, 905, 2347, 2209, 1419, 1040, 1068, 4314, 3735, 2375, 2214, 690, 3999, 2353, 2575, 988, 3057, 896, 238, 2877, 17, 3022, 3963, 2422, 4519, 1986, 2685, 259, 1933, 329, 2610, 1043, 15, 1451, 1068, 2130, 3203, 1055, 106, 4579, 3163, 3393, 4716, 50, 961, 588, 3868, 2457, 4122, 3715, 817, 25, 1718, 804, 533, 2683, 3951, 3551, 1865, 2927, 1169, 763, 232, 277, 1457, 2122, 3282, 3795, 4688, 2246, 2634, 2878, 4138, 1132, 2977, 444, 953, 784, 2180, 226, 1528, 1345, 4396, 2033, 1173, 1086, 4557, 4030, 3798, 2945, 2041, 3098, 1432, 4651, 653, 3951, 2756, 2000, 4941, 1469, 1811, 4996, 691, 747, 1939, 2943, 4830, 3266, 2458, 2925, 1377, 3126, 810, 3137, 1648, 3382, 2598, 2179, 451, 247, 276, 1943, 4380, 1532, 2563, 128, 1499, 2385, 1767, 2940, 2596, 383, 4299, 4386, 1071, 378, 1487, 26, 3705, 1946, 2210, 4996, 2841, 4575, 4913, 3987, 542, 2825, 696, 2627, 1311, 3630, 2544, 2865, 1810, 493, 962, 3954, 4609, 4858, 1523, 647, 3937, 43, 4601, 4655, 1477, 1315, 2341, 821, 868, 3117, 737, 3604, 1102, 339, 884, 4872, 2412, 2847, 1732, 134, 4985, ...snip... 179, 4357, 1777, 3304, 4713, 2374, 2754, 216, 4748, 2167, 91, 702, 1245, </pre>	252798	252798 ✓

Test	Expected	Got	
<pre> 1218, 4287, 729, 3777, 575, 1151, 878, 2870, 2520, 4635, 3721, 772, 4134, 1268, 4989, 1632, 3530, 4601, 3952, 399, 758, 1920, 3467, 2536, 4245, 392, 536, 3979, 3439, 148, 4012, 4577, 2554, 778, 288, 435, 3286, 1493, 290, 2328, 1871, 4825, 4090, 4513, 3441, 4661, 3738, 1768, 2327, 93, 660, 385, 1924, 1576, 987, 1759, 966, 711, 579, 825, 1511, 1312, 3866, 2527, 598, 1537, 1547, 3480, 2461, 4497, 2248, 869, 4348, 1740, 608, 1532, 2823, 4277, 10, 4061, 2076, 4297, 1529, 4321, 2655, 2841, 3304, 3789, 1505, 5000, 744, 4513, 2526, 2806, 2479, 1637, 1142, 1308, 2931, 4670, 1550, 241, 1936, 598, 4064, 613, 2382, 1010, 1224, 1476, 150, 2136, 4561, 53, 2302, 4776, 583, 1547, 3883, 1426, 4046, 4131, 1071, 3648, 459, 3127, 2556, 382, 4194, 2723, 1968, 775, 2368, 310, 3329, 546, 3809, 1260, 3154, 4909, 421, 4121, 1526, 1544, 3014, 4784, 3364, 3789, 3405, 4493, 1676, 2702, 989, 4096, 4010, 4080, 4888, 1739, 1664, 1927, 4514, 1301, 1361, 178, 4027, 4489, 3659, 2209, 56, 848, 1308, 3500, 20, 1046, 297, 301, 419, 3804, 1434, 2248, 1647, 2753, 2194, 2578, 3951, 3774, 1646, 504, 3220, 2378, 1788, 834, 6, 4380, 686, 4040, 3471, 2367, 3951, 4011, 1797, 4435, 2101, 367, 2462, 2585, 879, 583, 1208, 4775, 4481, 2122, 4207, 1133, 1438, 4637, 3226, 1631, 2942, 2625, 695, 4748, 4776, 4126, 3862, 2718, 2465, 1296, 1012, 3971, 2949, 769, 1705, 2195, 4269, 2407, 3556, 3347, 1482, 959, 725, 3250, 4041, 2618, 2275, 3178, 4736, 463, 1095, 4553, 4799, 1762, 385, 2288, 2427, 756, 2418, 3892, 3319, 2498, 2026, 1363, 4260, 1994, 1130, 3889, 115, 4291, 4172, 3372, 800, 228, 1834, 2133, 4593, 1792, 3181, 1704, 134, 801, 1823, 3477, 3764, 1475, 3730, 2712, 869, 1066, 3994, 1133, 2283, 438, 2489, 2577, 3677, 4662, 1714, 2701, 2120, 4003, 248, 4740, 4089, 4119, 3847, 4447, 4722, 3245, 3779, 2191, 2243, 18, 584, 4995, 483, 4172, 1698, 4698, 1719, 2866, 1127, 2259, 795, 1757, 3441, 4379, 1319, 4374, 471, 3631, 2403, 3376, 1087, 498, 4862, 1982, 2525, 3338, 3723, 3888, 2063, 462, 1157, 3250, 2308, 818, 1317, 2806, 1394, 663, 2617, 3660, 3335, 740, 2703, 1406, 1522, 2810, 2795, 4193, 1939, 4566, 1029, 1481, 1056, 2968, 422, 3939, 74, 2068, 2228, 1177, 1088, 6, 765, 1490, 602, 12, 1729, 3640, 4984, 4527, 530, 2342, 1051, 4633, 3728, 2114, 3972, 2109, 272, 4111, 2101, 3516, 4397, 4006, 2893, 436, 794, 3198, 3456, 1254, 3815, 3113, 1741, 3583, 3746, 961, 1057, 1759, 1823, 715, 2469, 1713, 2379, 4738, 4460, 4882, 4766, 2151, 328, 3558, 1943, 3281, 3389, 3607, 4084, 3879, 4397, 2159, 866, 3197, 3309, 4400, 3930, 1543, 2744, 457, 3269, 1511, 3483, 3132, 2417, 1917, 2217, 1382, 3876, 4136, 2424, 2015, 816, 872, 1260, 1635, 3720, 1085, 2450, 2552, 4074, 2877, 1200, 2201, 2709, 4306, 4491, 2362, 429, 3323, 3416, 860, 1293, 989, 3599, 787, 660, 144, 2125, 1431, 2292, 2982, 793, 4984, 2591, 3706, 4738, 2325, 3074, 4998, 150, 1487, 222, 1087, 3998, 57, 4956, 3821, 1072, 1115, 809, 1389, 3677, 4296, 3571, 865, 2413, 1500, 3883, 4372, 3991, 3490, 2619, 654, 1862, 2324, 2138, 3327, 4328, 348, 1269, 3257, 3800, 3363, 1329, 1374, 2391, 1729, 30, 428, 3574, 1756, 2742, 1454, 2975, 700, 1342, 78, 1309, 2702, 1284, 1887, 4145, 695, 4178, 1224, 2143, 2156, 2218, 1253, 4200, 2464, 2827, 496, 2696, 1790, 2530, 2681, 507, 255, 844, 817, 2268, 2733, 3003, 845, 2430, 2585, 4639, 1185, 1959, 3643, 7, 3291, 50, 2152, 1777, 154, 2363, 1024, 4158, 2319, 3584, 3633, 521, 3274, 4814, 2190, 1906, 3960, 3074, 1340, 4637, 299, 570, 918, 3544, 4131, 2155, 2895, 4518, 2516, 1947, 1993, 1301, 4192, 3697, 4003, 1247, 374, 4700, 2824, 535, 4130, 2314, 4782, 922, 80, 2883, 1180, 2176, 779, 420, 496, 834, 1115, 4416, 4463, 3727, 1959, 1164, 1078, 290, 486, 841, 2241, 1590, 4044, 3216, 594, 2213, 1739, 545, 1036, 4725, 2608, 285, 3880, 1620, 1864, 3309, 1656, 701, 1483, 2938, 1434, 3723, 2968, 3422, 2712, 4782, 521, 1306, 3086, 2828, 2279, 1890, 3995, 1937, 81, 1798, 539, 4786, 3266, 506, 2563, 815}; cout << mountainWithoutValley(nums); </pre>			
<pre> vector<int> nums {1985, 761, 4863, 3463, 436, 1717, 474, 2248, 2727, 3850, 2140, 311, 2490, 4188, 4592, 3117, 4857, 1725, 1374, 723, 3168, 860, 286, 3047, 3382, 467, 655, 3407, 3665, 1639, 1127, 2533, 2035, 2069, 3171, 3860, 604, 179, 769, 3458, 3943, 2517, 4332, 1399, 2705, 1994, 1384, 2013, 2248, 2166, 3831, 4664, 4192, 1664, 3196, 3031, 36, 3615, 2029, 3004, 3579, 752, 3334, 480, 3616, 3139, 2980, 707, 1334, 520, 4102, 1683, 644, 2828, 699, 3810, 4102, 1336, 2256, 2157, 3821, 739, 921, 290, 2248, 735, 459, 2410, 2210, 162, 23, 4528, 573, 2798, 913, 2664, 3127, 2962, 4969, 2923, 3524, 474, 4385, 2842, 1653, 2037, 969, 393, 1328, 1462, 4264, 3878, 1248, 622, 3540, 2034, 4462, 415, 2164, 3576, 2616, 2932, 427, 4955, 1377, 247, 711, 2269, 4745, 3899, 3019, 3221, 2230, 2531, 650, 800, 4106, 2466, 2534, 2251, 3728, 2533, 4048, 448, 4514, 53, 1856, 3656, 4356, 34, 3519, 4096, 4510, 4277, 538, 2286, 2950, 1484, 4162, 4350, 1492, 604, 818, 336, 2202, 2857, 1513, 4060, 2380, 2034, 361, 1513, 4804, 1691, 599, 1474, 4146, 2874, 4236, 3867, 2914, 3134, 4851, 1178, 4997, 4917, 4692, 3313, 3188, 2414, 3888, 1371, 1402, 3870, 2486, 4029, 2890, 4606, 4014, 1303, 3002, 4260, 2620, 4635, 526, 297, 4102, 4509, 3448, 1292, 3857, 171, 278, 1941, 4378, 3903, 1188, 636, 425, 315, 2668, 4851, 1026, 2669, 1355, 1785, 1415, 2315, 134, 2377, 4776, 2544, 1972, 2730, 3981, 256, 3254, 1160, 3253, 3976, 108, 2579, 2096, 1633, 754, 1198, 4898, 4295, 4043, 587, 1776, 4901, 3837, 4829, 3440, 2756, 3774, 269, 2500, 1467, 4, 4986, 544, 744, 1371, 3859, 85, 1880, 2274, 3534, 708, 1708, 1287, 1303, 4063, 37, 3877, 1561, 1564, 3481, 1763, 3984, 3755, 3630, 751, 362, 274, 1319, 3859, 3611, 1607, 4853, 2198, 464, 414, 1271, 2126, 2431, 844, 2965, 2108, 3661, 3707, 1409, 2992, 935, 4142, 1495, 576, 2902, 720, 534, 4163, 2876, 3408, 2391, 2243, 916, 1328, 1614, 658, 3929, 566, 792, 1107, 525, 4138, 3596, 2405, 2758, 4905, 3524, 900, 4764, 4074, 262, 4544, 4477, 2687, 4837, 3706, 4308, 2758, 3630, 3836, 4912, 984, 3096, 4072, 169, 1521, 379, 3228, 641, 3786, 1396, 1356, 1632, 3820, 2368, 4224, 3056, 1886, 4600, 4256, 3210, 4288, 3483, 270, 1065, 1552, 458, 819, 1853, 3189, 1120, 4452, 2803, 1003, 376, 3390, 483, 3126, 3452, 2240, 1529, cout << mountainWithoutValley(nums); </pre>	265566	265566	✓

Test	Expected	Got
<pre> 3339, 339, 656, 3929, 1147, 2410, 2376, 1069, 3441, 486, 4178, 2889, 1950, 2537, 1114, 2129, 3562, 2004, 3627, 289, 3182, 996, 4693, 2628, 3952, 3192, 350, 1737, 3535, 4081, 606, 2747, 2800, 1242, 4719, 2312, 696, 2115, 2732, 3445, 4427, 1505, 129, 1368, 2012, 2992, 3028, 2470, 1349, 1915, 1307, 3582, 1258, 3659, 2086, 2765, 1825, 1527, 3726, 4723, 3028, 2923, 86, 4438, 2787, 2596, 3417, 2589, 467, 1122, 1709, 824, 1204, 1767, 3502, 431, 4546, 55, 277, 1952, 2441, 3121, 4426, 4917, 3287, 1381, 4704, 493, 3116, 3086, 4761, 2217, 1469, 2506, 217, 1186, 4346, 71, 3551, 2553, 1972, 2640, 1265, 519, 1635, 3640, 4848, 3846, 1975, 4111, 3204, 503, 1942, 2875, 3249, 4462, 3029, 4377, 1162, 2083, 2290, 4900, 3316, 675, 1609, 3301, 1615, 2102, 566, 3667, 1080, 1442, 2915, 2821, 1674, 2136, 4554, 3286, 4415, 4061, 2381, 2415, 2698, 1956, 3155, 1156, 2734, 3297, 1598, 4200, 3388, 2713, 2547, 4490, 2257, 4073, 1322, 2116, 438, 4132, 2735, 4409, 4428, 3249, 3351, 4688, 4288, 3116, 3063, 118, 4379, 508, 658, 4632, 2421, 1901, 3121, 4497, 4235, 2255, 2456, 3113, 1623, 390, 1895, 935, 839, 2299, 830, 1472, 3392, 4819, 4180, 3809, 1180, 647, 690, 1441, 465, 1500, 3467, 4578, 1077, 1299, 2359, 602, 2119, 4836, 920, 2693, 1568, 3614, 2492, 2485, 3881, 418, 3237, 1036, 246, 4457, 478, 2993, 492, 75, 3663, 4938, 847, 4986, 226, 3153, 415, 926, 1878, 3130, 3308, 1530, 3156, 4268, 4848, 2551, 1064, 3790, 2707, 1104, 2642, 3859, 4261, 1469, 4286, 3689, 1359, 3868, 3497, 711, 2835, 340, 1982, 1357, 405, 2372, 4608, 2304, 4354, 4564, 2072, 1030, 1265, 3282, 1072, 4750, 1974, 1904, 105, 2396, 326, 435, 2266, 652, 168, 47, 3723, 2741, 1985, 834, 473, 3833, 1523, 2190, 1442, 953, 2687, 891, 1603, 3781, 4112, 4485, 2632, 4557, 2602, 4545, 1485, 3042, 4875, 885, ...snip... 3788, 791, 4478, 3247, 1002, 922, 284, 3504, 4216, 1603, 4177, 1580, 1947, 1097, 4190, 1314, 252, 4798, 2587, 1505, 646, 2054, 2180, 4227, 3541, 2945, 711, 4761, 4414, 1411, 4413, 488, 3817, 3994, 416, 3978, 1758, 3903, 2412, 4951, 3687, 4129, 2728, 2854, 698, 3854, 1647, 4815, 4219, 2180, 1850, 2971, 991, 2055, 1432, 286, 2387, 3883, 4417, 2787, 1262, 3125, 167, 736, 3156, 2877, 6, 3180, 2197, 2191, 118, 1335, 4581, 1546, 823, 705, 3469, 2135, 2996, 4622, 3083, 654, 1554, 2487, 1909, 4630, 251, 2564, 1203, 1780, 1630, 441, 4838, 4415, 682, 4095, 4379, 4065, 1659, 3821, 1517, 752, 1691, 359, 1354, 4213, 1645, 469, 212, 4900, 1246, 4903, 836, 3134, 1157, 3719, 2210, 1460, 628, 4887, 2247, 828, 199, 4606, 1086, 4569, 4589, 4469, 664, 1758, 2573, 2494, 4296, 2443, 4523, 4199, 758, 691, 4938, 1107, 1168, 3997, 4044, 2066, 2606, 1266, 2832, 4491, 1076, 1574, 892, 1087, 2837, 3955, 1227, 159, 702, 226, 4088, 1840, 250, 587, 972, 486, 2628, 2472, 3413, 1647, 1217, 1044, 935, 1810, 1951, 3625, 1983, 1904, 281, 971, 318, 1785, 213, 1359, 4546, 4378, 895, 3361, 4611, 70, 3599, 3684, 2093, 3970, 280, 4458, 569, 1013, 2291, 3398, 3337, 2040, 2916, 4072, 4196, 3699, 3466, 874, 4758, 2448, 1885, 4137, 3372, 2283, 2235, 2318, 2503, 218, 4940, 4981, 1271, 3074, 2831, 1285, 617, 2466, 1892, 3777, 4404, 4220, 1043, 3439, 3376, 4721, 1637, 79, 2000, 2052, 298, 182, 4474, 2810, 3274, 4156, 13, 2361, 2280, 3842, 3082, 2374, 3556, 2952, 382, 3763, 4512, 566, 1855, 1713, 125, 4921, 1004, 4946, 326, 393, 2786, 651, 4301, 337, 3188, 1311, 709, 3342, 643, 3183, 905, 3155, 2047, 2553, 3627, 4908, 940, 3463, 391, 1218, 2681, 4730, 2360, 3111, 2637, 1638, 83, 3002, 2690, 4423, 1331, 4949, 3230, 3564, 2357, 389, 2088, 1148, 2717, 2802, 184, 3766, 760, 2000, 4538, 325, 2076, 148, 1981, 1108, 2996, 1387, 1623, 4422, 3153, 1696, 4435, 1017, 2470, 2404, 2269, 4116, 1162, 147, 477, 510, 619, 1748, 4587, 2625, 785, 593, 4122, 840, 2728, 3057, 1343, 2836, 4339, 1546, 2908, 4013, 3453, 2642, 2224, 1519, 1809, 3144, 3898, 3635, 2518, 3479, 1878, 4112, 4443, 4147, 1007, 4081, 1201, 2430, 2061, 1504, 3592, 4405, 1424, 1227, 1226, 2431, 1806, 4175, 1880, 3438, 3370, 1426, 4441, 4601, 2758, 3862, 3002, 4936, 1778, 4354, 2131, 3082, 4141, 4731, 3942, 2996, 2634, 179, 117, 1955, 2447, 2704, 1268, 972, 2931, 2833, 1805, 2729, 4598, 2589, 2611, 3047, 3190, 908, 4241, 3382, 3055, 1217, 2851, 4673, 4002, 1353, 4973, 1806, 2705, 2685, 4947, 2327, 1121, 254, 145, 811, 1937, 3610, 924, 3223, 4658, 1337, 80, 297, 1112, 3254, 892, 2182, 3529, 562, 1195, 4378, 667, 819, 1156, 1768, 3906, 2813, 3612, 3979, 2242, 4383, 3658, 661, 1866, 4602, 2963, 989, 4978, 2997, 3352, 3670, 4494, 732, 4501, 2445, 2068, 1578, 3067, 1075, 3490, 1805, 2016, 1181, 4707, 753, 237, 4600, 1700, 1304, 144, 3201, 1217, 392, 3795, 1636, 2142, 757, 2549, 450, 2579, 2105, 4096, 2177, 2735, 1089, 3579, 759, 3136, 3826, 1392, 1774, 1818, 1596, 712, 1655, 311, 4877, 558, 56, 2765, 3865, 2265, 1271, 1350, 1513, 2222, 3476, 1628, 105, 710, 1962, 3467, 3566, 1844, 4263, 2084, 2488, 4729, 557, 2087, 3725, 3411, 1447, 1896, 743, 1892, 4631, 3102, 2902, 2871, 3983, 4360, 4152, 3808, 4263, 3152, 4400, 556, 2831, 2949, 1805, 2622, 3725, 3467, 2477, 572, 2265, 2118, 1090, 2129, 3663, 2175, 2467, 2045, 2917, 4518, 2294, 3530, 982, 264, 4773, 1072, 1278, 1147, 830, 363, 1983, 4856, 943, 4627, 931, 2296, 3866, 757, 443, 1123, 3093, 168, 1943, 2588, 1366, 3858, 103, 4913, 2101, 500, 1362, 2345, 378, 899, 2012, 7, 1824, 1041, 2493, 2966, 1410, 3660, 2976, 1789, 4958, 3948, 1361, 4342, 2071, 955, 555, 1002, 1599, 2504, 873, 119, 3332, 4756, 4171, 1140, 2674, 2897, 1540, 987, 2970, 2104, 3297, 1024, 2858, 642, 532, 3762, 1180, 489, 2290, 4768, 1651, 16, 593, 94, 2748, 1888, 4543, 4232, 1642, 3132, 2287, 4278, 316, 3567, 1669, 1648, 626, 2573, 3868, 156, 286, 1384, 4149, 3574, 296, 3652, 3419, 2655, 3741, 3244, 941, 633, 1464, 774, 219, 737, 537, 2292, 549, 2430, 494, 2357, 1769, 970}; cout << mountainWithoutValley(nums); </pre>		

Passed all tests! ✓

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 9

Chính xác

Điểm 1,00 của 1,00

Given an array `nums[]` of size `N` having distinct elements, the task is to find the next greater element for each element of the array. Next greater element of an element in the array is the nearest element on the right which is greater than the current element. If there does not exist a next greater of a element, the next greater element for it is `-1`.

Note: `iostream`, `stack` and `vector` are already included

Constraints:

`1 <= nums.length <= 10^5`

`0 <= nums[i] <= 10^9`

Example 1:

Input:

`nums = {15, 2, 4, 10}`

Output:

`{-1, 4, 10, -1}`

Example 2:

Input:

`nums = {1, 4, 6, 9, 6}`

Output:

`{4, 6, 9, -1, -1}`

For example:

Test	Input	Result
<pre>int N; cin >> N; vector<int> nums(N); for(int i = 0; i < N; i++) cin >> nums[i]; vector<int> greaterNums = nextGreater(nums); for(int i : greaterNums) cout << i << ' '; cout << '\n';</pre>	4 15 2 4 10	-1 4 10 -1
<pre>int N; cin >> N; vector<int> nums(N); for(int i = 0; i < N; i++) cin >> nums[i]; vector<int> greaterNums = nextGreater(nums); for(int i : greaterNums) cout << i << ' '; cout << '\n';</pre>	5 1 4 6 9 6	4 6 9 -1 -1

Answer: (penalty regime: 0, 0, 0, 5, 10, ... %)

Reset answer

```
1 // iostream, stack and vector are included
2 ✓ vector<int> nextGreater(vector<int>& arr){
3     int n=arr.size();
4     vector<int>ret;
5     ret.resize(arr.size());
6     stack<pair<int,int>> s;
7     s.push({arr[0],0});
8     for(int i=1;i<n;i++){
9         pair<int,int> temp=s.top();
10        if(temp.second < i){
```

```

11    while(temp.first<arr[i]){
12        ret[temp.second]=arr[i];
13        s.pop();
14        if(s.size()==0){
15            goto stop;
16        }
17        temp=s.top();
18    }
19    stop:
20    s.push({arr[i],i});
21}
22 else {
23    s.push({arr[i],i});
24}
25}
26 int stackn=s.size();
27 for(int i=0;i<stackn;i++){
28     pair<int,int> temp=s.top();
29     ret[temp.second]=-1;
30     s.pop();
31 }
32 return ret;
33 }

```

	Test	Input	Expected	Got	
✓	<pre> int N; cin >> N; vector<int> nums(N); for(int i = 0; i < N; i++) cin >> nums[i]; vector<int> greaterNums = nextGreater(nums); for(int i : greaterNums) cout << i << ' '; cout << '\n'; </pre>	4 15 2 4 10	-1 4 10 -1	-1 4 10 -1	✓
✓	<pre> int N; cin >> N; vector<int> nums(N); for(int i = 0; i < N; i++) cin >> nums[i]; vector<int> greaterNums = nextGreater(nums); for(int i : greaterNums) cout << i << ' '; cout << '\n'; </pre>	5 1 4 6 9 6	4 6 9 -1 -1	4 6 9 -1 -1	✓

Passed all tests! ✓

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 10

Chính xác

Điểm 1,00 của 1,00

Given a string consisting of 'a'-'z' and brackets '(' and ')'.

Your task is to implement a function with following prototype:

```
string parenthesesReversal(string s);
```

The function returns the string after recursively reversing every string enclosed between brackets.

Note:

- The `iostream`, `vector` and `stack` libraries have been included and `namespace std` is being used. No other libraries are allowed.
- You can write helper functions.

For example:

Test	Result
string s = "vi(ik)kq"; cout << parenthesesReversal(s);	vikikq

Answer: (penalty regime: 0 %)

Reset answer

```

1 stack<int> stac;
2 void rev(string& s, int start, int end);
3 string pararev(string &s, int i);
4 string parenthesesReversal(string s) {
5     // STUDENT ANSWER
6     return pararev(s, 0);
7 }
8 //void rev(string &s,int start,int end);
9 string pararev(string &s, int i) {
10    //cout << i << endl;
11    //cout << s[i] << endl;
12    //cout << stac.size()<<endl;
13    if (s[i] == '\0') {
14        return s;
15    }
16    if (s[i] == '(') {
17        //cout << "not ok" << endl;
18        stac.push(i);
19        s.erase(i, 1);
20        return pararev(s, i);
21    }
22    else if (s[i] == ')') {
23        rev(s, stac.top(), i - 1);
24        s.erase(i, 1);
25        stac.pop();
26        return pararev(s, i);
27    }
28    return pararev(s, i + 1);
29 }
30 void rev(string& s, int start, int end) {
31     for (int i = start, j = end; i < j; i++, j--) {
32         char temp = s[i];
33         s[i] = s[j];
34         s[j] = temp;
35     }
36 }
37 }
```

	Test
✓	string s = "vi(ik)kq"; cout << parenthesesReversal(s);
✓	string s = ""; cout << parenthesesReversal(s);
✓	string s = "dblcblpb(vgmoj(dbmvy))"; cout << parenthesesReversal(s);
✓	string s = "utjnlvnfdrwzxvetrsr(dntn)mokesvzavcrnulsclfzvgttkpwqumzytlub()t(hnzwpfrojjjjrlkxth)bpsdktktw(ivd)d"; cout << parenthesesReversal(s);
✓	string s = "jeadm(ikq)dghlv(q(p)okihuwshgzufaangetetu)aw(dvuunwr)(qvnvim(pxiiitvw)whcv)pb(amas((nar)h)t(tixs())uynmjeeeitpeejl cout << parenthesesReversal(s);
✓	string s = "gxulk1(kvpyplooms(tdogh(gvnpqjtkdvm)i)wnknyqa())pgxfqtldlwy(qlaxilsoacykjuouxseieowzxucekupig)sdwzgscfezmoxt(zbkdr (rei)jlchqtfmlufhprpmweznzhvudtcg)ihrqpdtx(qthvsyuuksc1cqklqbzberq(nbk)wntaresszfb(aecbmoqpgcc)tbqsm(knfxkcvhrner(zfch:((dwjkkmcwhzdprnmdm(cdinajevqqnibospvrlyvcfcfqytvubbdj1)mll)jvhgjtbtvinomrkltv(oujfwqdrq))loubjkpr(m)waccsituejunwlwckmj1c)); cout << parenthesesReversal(s);
✓	string s = "avnewzlns(ott(yvcq)(kruameggzg))hqshgobucz(h(ljsyafysejfwoz)fgu)eu(uz(ab)safbmvcobjyddkuvhw(nbnyfwplda(hdqumxjm1 (cdwchotexb(nbxqbnzdcgkr1tc(apzptocqekqvzbqeb1jrlkid(zdhftagd)(kwm)(upadyuwpqflygo)ugc)g(pftyrpwuwavlox)kiiruw(xpyjyig)g))\br...snip... lqw(jgailqlm(jxwsxwgqytdm)ryjhvcntcipdeeyhj)uplbgogh(pr)h(wokdxgxceggmy)noucxwacvydytz)lclnamwcokizwnhdjmzaudqc (iohemcmzzsglhiaginiylsokpkiquctxlrrscaimbl(lhvww)hx)yz))rrtddoajdetkluhxhtwsshmrliqdn(oxwd)mps)a(vw(imxbobhtw1)ypd)r(mi(rarli (yismimui)mumzoqop))kogfsdt)owbtndxgrdtrpasjwmqzvluav1jd))hfwhhzjyfjodmgkapxiwy(ccnshltzh)hgk(lycwugmhf)gkopdqukw)qy(el cout << parenthesesReversal(s);
✓	string s = "f(dfaglopguwyuseaeehxfeomldut)ork(yjy)cf(ndy(tky())ghgsmfwo(cxseqorej((xpiip)cj((oexd)fb)nnkeakxehfans(wedgicis1 (fsnzufbqsikpcsvmkqdqn1)wo(jibaxr(qeavdinlndj)mr(fgm(bfsqarpnjvbxjslhttieszfatsc)ubzxazjyeewwtbgy(obgwr(refynzjbe(ximrmw (qtkxh(moyhffuladmtwyvvz)(jkavlp(crnhqaloqvp(gkzbacqcmvhezbc)dbnlnbh(aokt)opzdswt)ydqeqglcnlhgvqstos(uwtrglgnqqt)eirwf; (kpgslmoeukaherjzz(ctpdevzdbqlahoekekzs(j(akrsar(jlhfwdxjzounopvf)vtc)niuoydjwdfoocoyqsnuowdbhgmkfmm)(j)eyrqxajetp(cn)c1wl (ox((xv1kmprij)iyinosoxkhipt)htxqfj)wcrvh(nwwepdabbn)yfczjznpkwo(v(t(iyl)qmhb)jclt(aeiwmwffhb((h)weijcoejecdqxd)hieapryegtb cout << parenthesesReversal(s);
✓	string s = "oggypep(q(he)idvxlooqwjedsiuioq)iuskdkdey(hikk(kfr)a(c)gizubxmwigoto)ymwfaen((awbcgqkemefwhpgzh)cd())jjznff(nsrggd: (fqxmkcwex)suo)owgqmwdfglupfbknguov(oflvpadffstvssap)s)zofzbuaihikzyimjcoycamjv(kqdgfsxx1tmfzotwom(fpzj)bnkbraf(j)jlmebeqwhft: (aasbidnn)br1wmzjdukerqomwrkssnrofl(by)llyudkgmlembmqskvtcpf(cofaajaukzjaratuv(nlnqys1ndmv)ubfaf(kwlrdfnkatee)pda)j()kqrw: (fzcv(knhsjmbaxaz)q((eawuvdgfoqkwqotq)w)mgu)))nyrlrbqa)lcxpiopnynjprswu)rpvy(wgzbr(glufdc(bp)meolpzxytovdypjpmuhkslsyvm): (xh(wu(jriksm)xnqceavjfnejibqn1)mxor((gtobzsscudeg)))jshranxmtdm(ewoypwxkephdylt)wfsuernr(qdneotwmuewjf(jaxdjohhje)z((pcnv cout << parenthesesReversal(s);
✓	string s = "bpwn(l(s(pn)j)gsutvyoo(xbvvfntomljdr(cyffyaw)mkwjda)xgpkytmfz1qdqd)tzsgfjfjrrct(ijrvqg)wulqihxhxd(je(hhmsd(xi (podxmedkmbxmrpnzkwvbonaxucbgznktaucpqg)gta((nmak)vdpffiqjjccsndnrhethuaxggdyfoieprnxjld(aechirof(ubnodmhz)r)cpjyphnfmqbr (wyiyvbgyyxsyj)wq)q)pqbxgd(vw)stthdpuut(qlxioflkrzibkosniznmyqq(whbug)xkznmmi))jnby((maabeehq(qfp(u(wpedtccgkcgny(dk)zpnme (ao1jd)tdfjbjxqwrriyisd1spphyswnlwqfmag)kt)kukbpgaeurglkdyonzxzrawdrr(ykh(ytvynztcjzosdxskuijwbicpsrsxfwrjqascbdpyvg1acggb)q) (ytihloaqos)uejmv)czapxocogzqghxhsqaqfowzjlh)(szbgozboe)mnnjuybqzaddomppcjavjtalxpfkxrygxfpgm)uawgoqr))otvkircds()yxiv: cout << parenthesesReversal(s);

◀ ▶

Passed all tests! ✓

Chỉnh xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 11

Chính xác

Điểm 1,00 của 1,00

Given a string **S** of characters, a *duplicate removal* consists of choosing two adjacent and equal letters, and removing them.

We repeatedly make duplicate removals on **S** until we no longer can.

Return the final string after all such duplicate removals have been made.

For example:

Test	Result
cout << removeDuplicates("abbaca");	ca
cout << removeDuplicates("aab");	b

Answer: (penalty regime: 0, 0, 0, 5, 10, ... %)

[Reset answer](#)

```

1 #include<string>
2 string removeDuplicates(string S){
3     /*TODO*/
4     stack<char> stac;
5     stac.push(S[0]);
6     for(unsigned int i=1;i<S.size();i++){
7         if(stac.top()==S[i]){
8             while(stac.top()==S[i]){
9                 S.erase(i-1,2);
10                i--;
11                //cout<<S[i]<<endl;
12                //cout<<stac.top()<<endl;
13                stac.pop();
14                if(stac.size()==0){
15                    break;
16                }
17            }
18            stac.push(S[i]);
19        } else {
20            stac.push(S[i]);
21        }
22    }
23    return S;
24 }
```



	Test	Expected	Got	
✓	cout << removeDuplicates("abbaca");	ca	ca	✓
✓	cout << removeDuplicates("aab");	b	b	✓

Passed all tests! ✓

[Chỉnh xác](#)

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 12

Chính xác

Điểm 1,00 của 1,00

Given a string s and an integer k ($k \geq 1$). Manipulate the string following these rules:

- Repeatly delete the earliest k-consecutive characters appear in string s.
- There is no k-consecutive characters in the final string (after manipulating).

Request: Implement function

```
string deleteDuplicate(string s, int k);
```

To return the final string.

Example:

- `deleteDuplicate("aabbbdddadd", 3).`
- The final string returned is `dd`.
- Explanation: Firstly, delete the "bbb" to get "aadddadd", then delete "ddd" to get "aaadd", then delete "aaa" and final string is "dd".

Note: In this exercise, libraries `iostream`, `stack`, `utility` and `using namespace std` have been used. You can add other functions; however, you are not allowed to add other libraries.

For example:

Test	Result
<pre>string s = "aabbbdddadd"; int k = 3; cout << deleteDuplicate(s, k);</pre>	dd
<pre>string s = "aaabbbbccddddeffffghiiiihhhhggggffda"; int k = 5; cout << deleteDuplicate(s, k);</pre>	aaabbbbccddddea

Answer: (penalty regime: 0 %)

Reset answer

```

1 ✓ string deleteDuplicate(string S, int k) {
2     if(k==0||k==1) return "";
3     stack<pair<char,int>> stac;
4     stac.push({S[0],1});
5     for(unsigned int i=1;i<S.size();i++){
6         pair<char,int> temp=stac.top();
7         //cout<<temp.second<<endl;
8         //cout<<temp.first<<endl;
9         if(temp.first==S[i]){
10             while(temp.first==S[i]&&temp.second==k-1){
11                 for(int j=0;j<k-1;j++){
12                     stac.pop();
13                 }
14                 S.erase(i-k+1,k);
15                 i=i-k+1;
16             }
17             if(stac.size()==0){
18                 break;
19             }
20             temp=stac.top();
21             if(temp.first==S[i]) stac.push({S[i],temp.second+1});
22             else stac.push({S[i],1});
23         }
24         else {
25             stac.push({S[i],1});
26         }
27     }
28 }
```

29 | }

	Test	Expected	Got	
✓	string s = "aabbbdddadd"; int k = 3; cout << deleteDuplicate(s, k);	dd	dd	✓
✓	string s = "aaabbbbccddddefffghiiiihhhhgggffda"; int k = 5; cout << deleteDuplicate(s, k);	aaabbbbccdddeda	aaabbbbccdddeda	✓

Passed all tests! ✓

Chính xác

Điểm cho bài nộp này: 1,00/1,00.

Câu hỏi 13

Chính xác

Điểm 1,00 của 1,00

Given an array of integers.

Your task is to implement a function with following prototype:

```
int sumOfMinSubarray(vector<int>& nums);
```

The function returns the sum of the minimum value of every subarray of `nums`. The sum may be too large, so the result should mod `10000`.

Note:

- The `iostream`, `vector` and `stack` libraries have been included and `namespace std` is being used. No other libraries are allowed.
- You can write helper functions.

For example:

Test	Result
<code>vector<int> nums {1, 2, 4, 3}; cout << sumOfMinSubarray(nums);</code>	20

Answer: (penalty regime: 0 %)

Reset answer

```

1 int sumOfMinSubarray(vector<int>& a) {
2     // STUDENT ANSWER
3     int mod = 10000;
4     stack<int>st;
5     int n = a.size();
6     int dp[n];
7     for (int i =0; i<n; i++)
8     {
9         while (!st.empty()&&a[st.top()]>=a[i])
10        {
11            st.pop();
12        }
13        if (!st.empty())
14        {
15            int pre = st.top();
16            dp[i] = dp[pre]+(i-pre)*a[i];
17        }
18        else {
19            // or it doesn't exist, in this case the current element
20            // contributes with all subarrays ending at i
21            dp[i] = (i + 1) * a[i];
22        }
23        // push the current index
24        st.push(i);
25    }
26    int res = 0;
27    for (int i =0; i<n; i++)
28    {
29        res+=dp[i];
30        res%=mod;
31    }
32    return res;
33 }
```

	Test	Expected	Got	
✓	vector<int> nums {1, 2, 4, 3}; cout << sumOfMinSubarray(nums);	20	20	✓
✓	vector<int> nums {}; cout << sumOfMinSubarray(nums);	0	0	✓
✓	vector<int> nums {9830, 5335, 3573, 9754, 1778, 2960, 4386, 6461, 2524, 4734, 4263, 2571, 4628, 1981, 19, 4444, 5446, 3117, 102, 6044}; cout << sumOfMinSubarray(nums);	8583	8583	✓
✓	vector<int> nums {4921, 3462, 7784, 7764, 4246, 1659, 3629, 1594, 7885, 2625, 2038, 3252, 597, 7346, 4043, 1762, 2867, 8365, 7464, 461, 4238, 3441, 5814, 4218, 9008, 6826, 1084, 1956, 7199, 7882, 4550, 3241, 780, 2476, 7972, 7699, 203, 8101, 9334, 2539, 789, 2652, 1462, 6745, 619, 3147, 9558, 3230, 7339, 5126, 2589, 7669, 2584, 540, 4380, 8695, 9893, 3164, 2949, 159, 2082, 6138, 1582, 8605, 2682, 6193, 6802, 4079, 2492, 4091, 9689, 4733, 9881, 7519, 7658, 8803, 8925, 6603, 3950, 7012, 1426, 9346, 8566, 477, 6433, 8231, 2175, 9241, 1914, 9755, 4133, 1683, 2535, 3220, 7222, 9488, 1227, 3150, 697, 5822}; cout << sumOfMinSubarray(nums);	4631	4631	✓
✓	vector<int> nums {2544, 5222, 2719, 600, 5010, 8197, 3144, 8364, 4134, 3449, 2665, 3262, 9024, 922, 6669, 8798, 5315, 1660, 7497, 4695, 3977, 2544, 84, 4538, 1653, 6336, 8039, 1611, 7447, 1835, 3010, 2761, 7985, 3421, 8687, 5396, 1368, 9443, 1834, 3578, 8285, 9598, 1697, 499, 1068, 8006, 7865, 6222, 1108, 3376, 9185, 9499, 3720, 9571, 1483, 1482, 2118, 1799, 3912, 6292, 3878, 2831, 9555, 5017, 7717, 6329, 2241, 9977, 4579, 4276, 104, 9047, 233, 5601, 2045, 4539, 7231, 5309, 1746, 8327, 3581, 7806, 3856, 924, 3056, 1595, 401, 8957, 7264, 4401, 8034, 2148, 7328, 2211, 2006, 3620, 4069, 690, 5806, 8032, 9002, 6734, 5720, 8586, 2259, 9419, 6326, 2334, 5921, 3032, 3137, 7135, 9318, 6780, 9741, 9074, 4329, 1473, 3070, 5386, 12, 8037, 4139, 628, 572, 4303, 35, 8820, 1721, 655, 6813, 6079, 6039, 7417, 1684, 5227, 691, 884, 2163, 5363, 4282, 5166, 1021, 8352, 1095, 6770, 4544, 4117, 4155, 9244, 955, 8409, 8402, 8476, 9277, 4314, 5123, 7226, 1936, 80, 7704, 7785, 7079, 3471, 288, 9646, 2387, 5682, 2513, 7235, 492, 9012, 3729, 3959, 8542, 5200, 2967, 1657, 354, 125, 3422, 7045, 4143, 2114, 5594, 4430, 8272, 5856, 3997, 2892, 4007, 6079, 6940, 8279, 658, 7929, 7854, 3301, 1918, 1415, 2597, 8657, 1097, 424, 6112, 2685, 4077, 4585, 7951, 2729, 4265, 162, 6217, 2500, 5186, 2573, 9867, 2017, 7562, 5196, 8946, 1470, 8399, 1947, 9467, 2978, 456, 1700, 6344, 9285, 5756, 338, 2630, 8901, 6003, 3416, 3085, 1174, 4624, 9858, 3112, 377, 6688, 1196, 5365, 1488, 6332, 4002, 7390, 928, 5829, 3416, 5994, 1384, 2497, 2027, 2142, 129, 7152, 5399, 9336, 3991, 7381, 6811, 7725, 7432, 8849, 3562, 2273, 1978, 2418, 6355, 6167, 3473, 1340, 259, 730, 9217, 1222, 905, 3916, 4865, 601, 4426, 1596, 7617, 5076, 2647, 781, 8157, 4739, 5244, 4676, 371, 6129, 4410, 7042, 5585, 8284, 4807, 7268, 1442, 5544, 8628, 4423, 4459, 8023, 9983, 8144, 6384, 423, 4234, 315, 5565, 9948, 2204, 2017, 642, 4608, 3815, 2480, 4590, 520, 282, 6182, 2399, 3801, 5927, 3165, 2105, 6267, 6695, 3166, 5238, 92, 2141, 3071, 8716, 8776, 3064, 9406, 4709, 6968, 5853, 8115, 6011, 2488, 789, 3953, 7490, 9261, 7523, 4990, 4363, 2343, 6856, 9172, 6774, 4660, 3370, 3798, 375, 838, 6067, 2102, 6221, 9077, 2646, 5190, 735, 4487, 5031, 590, 6935, 8047, 2379, 2894, 2756, 7380, 893, 3908, 682, 1364, 6675, 1184, 2589, 4137, 2448, 7702, 5054, 8826, 5196, 3460, 8421, 581, 897, 3877, 6575, 1372, 1271, 677, 3482, 9357, 9778, 6502, 6831, 7782, 6874, 9186, 5295, 2633, 2220, 6050, 3762, 7404, 4463, 9054, 6251, 7605, 4601, 4566, 5984, 1652, 1843, 7329, 4440, 1857, 3690, 8099, 6742, 6615, 9432, 7316, 725, 8129, 1267, 687, 1956, 7221, 4387, 40, 7765, 1262, 5610, 7122, 6833, 1870, 2223, 6176, 6974, 7763, 3691, 9594, 7635, 5417, 4664, 5699, 1274, 2641, 855, 4808, 951, 3073, 2225, 3547, 4713, 6091, 3520, 5425, 5359, 7106, 3647, 7464, 1709, 5131, 1967, 1534, 8255, 2337, 8811, 782, 6557, 2939, 5694, 284, 9103, 3793, 2532, 8451, 867, 381, 559, 3852, 1693, 1471, 8427, 4626, 4030, 1844, 4467}; cout << sumOfMinSubarray(nums);	4396	4396	✓
✓	vector<int> nums {7268, 9416, 2343, 2597, 282, 544, 1112, 563, 732, 4645, 4381, 1228, 688, 1441, 2006, 6651, 1170, 2206, 5685, 7998, 8888, 3642, 9678, 7641, 8278, 9710, 1163, 1564, 3696, 203, 4531, 3795, 8735, 6454, 7532, 8396, 2438, 5094, 4926, 7966, 4901, 3643, 4970, 1224, 4365, 5473, 7147, 8200, 5525, 854, 8620, 2829, 4364, 6360, 1834, 5935, 2640, 5074, 2268, 1613, 9022, 9401, 2144, 2327, 1083, 8130, 1431, 190, 2995, 3590, 7300, 9112, 2961, 2216, 769, 9769, 2638, 1606, 3859, 7582, 8124, 1679, 5358, 5243, 2885, 5957, 2540, 3176, 7456, 578, 762, 7767, 1221, 6895, 4916, 953, 7072, 1587, 5266, 8034, 1675, 1673, 578, 6234, 5335, 4517, 8137, 4277, 5915, 5736, 8180, 2101, 9713, 5106, 2820, 2671, 348, 1240, 2489, 9932, 287, 5574, 2318, 8432, 8794, 6438, 1925, 6160, 8198, 3298, 8884, 9503, 2244, 6854, 9345, 5466, 5648, 6794, 2058, 8614, 5260, 8398, 6696, 1894, 2538, 2582, 2085, 7297, 2233, 8425, 5691, 4066, 1893, 4901, 2303, 2930, 2600, 5349, 5566, 4500, 229, 2454, 7366, 319, 1895, 4054, 9800, 4706, 4565, 781, 5517, 1394, 6856, 4769, 2071, 2695, 7584, 9349, 1161, 9448, 6372, 6673, 5001, 1812, 7305, 7192, 3013, 868, 1382, 3893, 3236, 8125, 2133, 9272, 9951, 3721, 7403, 5600, 383, 216, 1484, 2009, 121, 2147, 6473, 4664, 4743, 3074, 7339, 3364, 6056, 6974, 1253, 8712, 5516, 9465, 7525, 4870, 3672, 2765, 699, 6974, 1185, 2019, 1613, 7759, 2807, 6629, 6202, 8461, 1151, 6630, 1773, 4260, 7174, 2384, 7765, 8076, 4002, 9965, 3027, 4685, 3099, 8634, 7715, 5749, 1179, 1690, 5055, 8616, 4897, 1127, 2304, 4135, 8972, 6766, 6182, 5716, 9030, 5179, 2538, 4245, 7167, 5874, 2306,	8084	8084	✓

Test	Expected	Got
5111, 444, 2833, 2277, 6868, 4144, 2066, 4608, 3320, 2153, 5173, 1720, 2561, 6478, 445, 3904, 7365, 5304, 9369, 1997, 2329, 8398, 3014, 3784, 9199, 1413, 4723, 8548, 4735, 9575, 2710, 1457, 393, 9556, 3437, 5601, 313, 5543, 1976, 3545, 5837, 6171, 9317, 1578, 2403, 6445, 8827, 1280, 3846, 2131, 8427, 9002, 3147, 2877, 7011, 3867, 9751, 4139, 7062, 4256, 3587, 64, 7497, 6340, 9731, 4901, 7822, 2289, 5091, 9121, 2358, 7271, 2595, 6104, 4707, 3489, 4868, 2193, 7228, 1430, 4579, 2316, 5685, 9358, 5590, 9139, 7838, 7383, 8245, 6625, 8746, 2038, 1867, 6401, 3427, 33, 1004, 2728, 4363, 940, 2521, 9109, 5942, 258, 9442, 4296, 9735, 8775, 9724, 9841, 8873, 7340, 1571, 9722, 2207, 9763, 6065, 6152, 173, 4019, 1048, 9677, 4472, 6622, 8044, 4130, 2059, 2657, 349, 1748, 2348, 811, 5796, 3052, 6591, 7410, 3251, 2787, 6474, 8339, 1378, 4357, 2584, 5192, 623, 7910, 3809, 6495, 85, 1730, 5993, 2200, 1057, 1033, 3688, 623, 2465, 9340, 2251, 4763, 5997, 369, 8586, 7974, 1051, 9037, 2697, 671, 7454, 7778, 1746, 972, 1646, 1254, 2786, 979, 5790, 4598, 2755, 6491, 6106, 9771, 8411, 1854, 6703, 893, 7827, 1778, 2880, 4197, 2922, 3945, 3365, 5623, 8027, 4693, 9063, 2928, 3788, 720, 970, 7932, 520, 3355, 4218, 6587, 6736, 8357, 7689, 7204, 2221, 6271, 7223, 9179, 602, 1645, 7411, 3616, 9231, 7407, 7217, 4184, 9664, 6529, 9965, 2543, 5912, 48, 2479, 4973, 9456, 1438, 6512, 537, 821, 1752, 5723, 2492, 1753, 790, 2423, 7304, 5672, 242, 8404, 2168, 3037, 8973, 6404, 4953, 9462, 8254, 5277, 5322, 7062, 4812, 6028, 2016, 2857, 5471, 1356, 2125, 9357, 3745, 2279, 882, 3280, 315, 1900, 8882, 7988, 7709, 9889, 3401, 250, 7934, 4795, 7036, 2862, 5509, 5164, 8659, 4831, 9111, 4268, 1413, 3602, 5362, 5579, 2930, 2535, 8585, 5925, 8487, 1516, 7297, 8568, 2878, 1584, 5547, 1426, 8980, 3963, 52, 1007, 6463, 6780, 8457, 8945, 9855, 8980, 3359, 1728, 9875, 5329, 8193, 6037, 5104, 5069, 715, 6433, 5566, 1224, 8295, 5523, 7636, 9555, 7718, 5626, 133, 2031, 9074, 9320, 9322, 1029, 1953, 4952, 8850, 3607, 1580, 2513, 2775, 1430, 7201, 9626, 7907, 166, 3301, 7073, 6819, 2847, 7678, 4663, 4363, 5494, 4161, 9662, 4762, 1627, 2135, 7494, 5952, 6740, 406, 6308, 6989, 602, 4589, 1924, 5387, 7217, 6655, 97, 8494, 4782, 6612, 642, 4523, 7765, 2892, 4039, 3063, 1153, 3493, 4596, 1761, 5591, 6399, 5261, 67, 7021, 6414, 4772, 565, 3206, 8640, 2222, 5139, 6929, 7872, 1452, 2289, 1620, 922, 8169, 7483, 7507, 3743, 5887, 5649 ...snip... 670, 7762, 7544, 4355, 1273, 2469, 357, 4524, 8147, 7692, 2760, 6750, 4185, 8002, 1741, 1751, 2325, 1418, 8062, 4715, 5081, 3440, 6772, 971, 1585, 9568, 2015, 7193, 1512, 2535, 1160, 9707, 2334, 8487, 757, 2590, 7816, 5758, 1773, 796, 9576, 9656, 2941, 2241, 863, 7092, 9050, 9611, 976, 3845, 9623, 5216, 5655, 9370, 1593, 6437, 4618, 5952, 7401, 2748, 2031, 3095, 5082, 4653, 6798, 2882, 8189, 3992, 3992, 6131, 1217, 4073, 4136, 9974, 6078, 349, 6296, 7948, 1428, 2853, 8936, 3675, 8079, 4953, 907, 3044, 2942, 2415, 7346, 3398, 8100, 2412, 6685, 3377, 5150, 3081, 3982, 5773, 1415, 954, 4657, 9980, 3241, 3803, 1050, 4077, 6836, 4203, 5981, 9746, 2308, 1070, 7315, 6837, 9373, 7678, 3308, 1846, 6858, 2370, 6496, 984, 5064, 5985, 2122, 9933, 6508, 8824, 1655, 5887, 9052, 8587, 1595, 145, 3676, 1305, 9014, 4693, 9687, 4115, 7557, 8111, 5777, 9566, 5003, 6553, 9259, 5398, 6722, 9608, 7271, 2709, 3804, 7855, 7980, 5365, 6383, 7459, 3884, 2050, 5361, 6480, 5684, 5796, 751, 1450, 1779, 9771, 5572, 149, 2303, 5260, 7769, 5113, 2741, 7212, 4200, 1648, 6687, 3914, 9173, 6610, 5995, 4481, 7087, 4292, 8041, 7933, 2739, 5404, 9893, 1407, 6722, 1944, 166, 5276, 4996, 5167, 1926, 5339, 8813, 9911, 3860, 2766, 5503, 7577, 1990, 8592, 3184, 3779, 2114, 49, 1150, 5116, 4431, 3374, 7472, 7869, 2407, 7636, 5854, 5198, 9504, 1877, 2372, 1199, 4556, 748, 2435, 1784, 5309, 8782, 136, 2434, 2269, 1091, 9425, 3240, 7437, 163, 8013, 200, 7500, 4420, 7300, 6607, 1867, 9715, 3863, 1764, 9742, 1136, 89, 9187, 5573, 1391, 7790, 7069, 2412, 1420, 9795, 3773, 6544, 9983, 1486, 79, 9715, 176, 6823, 1820, 8114, 3021, 8055, 1001, 773, 362, 1967, 9009, 2710, 1240, 1245, 1983, 7446, 4425, 9499, 1697, 548, 6979, 9001, 9674, 4335, 291, 3500, 1751, 7063, 1764, 8118, 8036, 1582, 992, 2983, 5023, 5405, 5669, 3220, 6333, 8596, 7772, 9192, 6699, 954, 5862, 5145, 7368, 7190, 8984, 9532, 3506, 7501, 8617, 2571, 9198, 8589, 6922, 6902, 9388, 1721, 4357, 5832, 1562, 4758, 1064, 1819, 8231, 1349, 4020, 9502, 1947, 4675, 7569, 8870, 6038, 4013, 597, 4403, 2358, 6616, 85, 5333, 141, 9135, 8155, 1150, 6961, 478, 824, 3050, 2966, 2191, 7099, 3740, 8745, 7635, 643, 9706, 3278, 2036, 8626, 669, 1965, 8680, 8930, 1666, 9832, 8888, 9868, 1172, 6993, 9330, 8487, 1741, 5273, 6012, 2078, 2394, 4348, 6810, 3606, 9151, 1819, 801, 9502, 7035, 3273, 109, 6535, 4413, 9515, 6976, 4678, 6567, 5034, 3484, 7591, 6866, 107, 2163, 5508, 2611, 5841, 5981, 4279, 524, 96, 3818, 7053, 2806, 2805, 360, 9595, 9341, 661, 9463, 5717, 3148, 4435, 8043, 3804, 6171, 1670, 802, 4776, 7590, 938, 1767, 2007, 6224, 8101, 6614, 6757, 7496, 4605, 416, 268, 2075, 2249, 593, 1613, 9434, 4917, 9873, 6912, 7453, 4490, 7976, 7305, 1127, 1581, 6038, 4076, 2140, 4759, 4962, 8929, 7441, 6757, 2176, 9963, 115, 1504, 9064, 7494, 9611, 9055, 3525, 1359, 7552, 284, 2267, 4473, 2638, 565, 1133, 4701, 4246, 9220, 1262, 465, 565, 6897, 2966, 1692, 3641, 7478, 4327, 1785, 7601, 5171, 6561, 4730, 9684, 9987, 6994, 6239, 1589, 3326, 6626, 6872, 3974, 3041, 1583, 3447, 7787, 5227, 8660, 5897, 7408, 9978, 3823, 6038, 8829, 8165, 6435, 7558, 1688, 7739, 3355, 9863, 5628, 4460, 9344, 605, 9162, 4134, 3005, 8921, 3837, 6083, 848, 3758, 818, 272, 7476, 2063, 1863, 4449, 173, 8924, 8787, 3876, 3588, 5039, 9367, 8771, 7252, 834, 1460, 7134, 4388, 9860, 2397, 5736, 2614, 2908, 781, 2048, 2047, 6108, 7308, 9892, 7751, 8921, 7262, 945, 5699, 186, 9629, 2142, 434, 6000, 9928, 1705, 5301, 357, 6172, 6626, 3267, 1704, 594, 2007, 9355, 8647, 9321, 2943, 5499, 3518, 5398, 5515, 1424, 5775, 9032, 1850, 3167, 5826, 1105, 3244, 7124, 624, 9154, 5050, 169, 362, 2070, 9011, 1594, 5056, 6547, 6314, 1062, 4142, 8124, 7918, 8072, 9433, 8362, 90, 3094, 7035, 9099, 5534, 6330, 539, 9356, 7421, 4432, 6193, 8147, 7338, 511, 824, 9056, 2748, 1611, 6277, 7846, 2249, 218, 1303, 1108, 1410, 5958, 383, 2777, 1339, 8242, 5602, 4791, 9551, 2001, 4620, 9367, 9553, 9843, 424, 3629, 2134, 4849, 2765, 45, 1537, 9739, 7339, 8494,		

Test		Expected	Got	
2110}; cout << sumOfMinSubarray(nums);				
✓ vector<int> nums {2960, 9031, 2351, 9859, 4046, 1442, 5479, 2164, 9243, 3681, 2977, 3495, 4387, 713, 9960, 3767, 7554, 5689, 3375, 7004, 1860, 93, 6053, 9883, 9227, 9311, 6454, 7210, 8274, 9664, 5477, 1143, 6688, 4936, 4841, 2266, 5534, 5227, 6901, 6119, 9244, 8317, 1309, 4824, 7217, 198, 230, 3547, 6594, 3449, 5229, 3768, 2933, 1618, 1371, 4630, 1110, 622, 5655, 1102, 7466, 4704, 5788, 3455, 4002, 8561, 1500, 804, 562, 7047, 389, 883, 4723, 2362, 7284, 6680, 8725, 4131, 8040, 6993, 1929, 9365, 3192, 4939, 8734, 1712, 7639, 3165, 5546, 5283, 1158, 7475, 3788, 1235, 5755, 4414, 2080, 8266, 977, 1562, 1826, 5937, 4272, 2432, 4246, 9376, 7342, 7933, 4327, 9989, 1910, 7811, 639, 3156, 7992, 4830, 1751, 8959, 690, 7882, 3551, 134, 2333, 1507, 8832, 2672, 7786, 6060, 2368, 666, 543, 9844, 2678, 8335, 7421, 6806, 6814, 6713, 7620, 8369, 4559, 2663, 2601, 5304, 5771, 3871, 4518, 1434, 663, 7654, 2434, 1235, 412, 3038, 9316, 3124, 6542, 9839, 2066, 4962, 5051, 1706, 9934, 9099, 6053, 5321, 2020, 1182, 1235, 8283, 1181, 2522, 9541, 1678, 2488, 1607, 1879, 3278, 7353, 7493, 968, 1198, 5330, 6818, 7387, 116, 8097, 5936, 5981, 3737, 4858, 3286, 9048, 4513, 5815, 5829, 2776, 5384, 797, 4494, 3981, 7950, 5249, 8610, 7280, 1205, 1723, 8922, 2249, 6936, 2473, 9432, 6048, 398, 5143, 9982, 7234, 4047, 7134, 8302, 6870, 7238, 2971, 8801, 2017, 7080, 4545, 1916, 6023, 1652, 2561, 9583, 1331, 6139, 9513, 6832, 8137, 2818, 3770, 5048, 8906, 4197, 5551, 5660, 1639, 703, 955, 9983, 4218, 5768, 1810, 9494, 2977, 621, 9291, 9340, 348, 4582, 1108, 6758, 6858, 4660, 6663, 6586, 2767, 9834, 2284, 1068, 360, 5501, 2638, 617, 914, 2671, 7490, 1032, 5831, 4968, 9621, 605, 8570, 7238, 7206, 767, 1494, 3271, 8948, 1101, 4706, 7060, 8489, 6205, 8696, 9585, 231, 1237, 2039, 3202, 7106, 9950, 7317, 112, 8414, 4629, 5000, 987, 2282, 9619, 135, 3458, 6846, 2304, 2028, 8674, 9793, 2595, 9197, 3052, 8686, 7581, 1723, 4506, 5527, 9784, 75, 6242, 6474, 9338, 8868, 8043, 1683, 5790, 1959, 448, 8963, 3380, 4934, 6036, 1920, 4711, 5646, 8309, 573, 1760, 5062, 6851, 2971, 4665, 749, 6930, 5791, 645, 2303, 3471, 1830, 9006, 4649, 4624, 1911, 5055, 7245, 575, 648, 2884, 5128, 6647, 5460, 8766, 5986, 1263, 2555, 3275, 5813, 1832, 3130, 2325, 9650, 4034, 8140, 345, 3168, 3748, 5329, 2220, 7782, 1531, 95, 4601, 8427, 9362, 6374, 5390, 4364, 1875, 8328, 9208, 2782, 1789, 5161, 2747, 9397, 5607, 7267, 7031, 9204, 8212, 1853, 4647, 6502, 6279, 9789, 7756, 1356, 5288, 1212, 9833, 9303, 1751, 8249, 6512, 8073, 4193, 4704, 174, 4455, 2978, 3212, 1954, 6002, 3594, 330, 8419, 5143, 4562, 3297, 4147, 5043, 1808, 4623, 4974, 5066, 665, 7637, 5928, 9802, 5207, 2358, 6614, 1212, 3755, 1148, 1131, 7067, 8565, 3356, 6534, 4951, 7010, 715, 5834, 988, 4420, 4848, 6175, 4703, 1880, 347, 9239, 2137, 1621, 5399, 5155, 9064, 5297, 5565, 1414, 3650, 7336, 3769, 218, 2123, 959, 576, 1298, 2312, 3797, 1026, 5504, 7556, 4283, 3181, 936, 3524, 4038, 2066, 9237, 8295, 6803, 2284, 6139, 1648, 1290, 8685, 1564, 2646, 9302, 9915, 8910, 7627, 4612, 3756, 4973, 6620, 6729, 2478, 1837, 4761, 5849, 8027, 2643, 9942, 9974, 5418, 2101, 49, 4086, 2488, 1966, 405, 664, 7129, 7895, 5390, 6328, 2298, 7250, 7006, 7297, 8425, 3401, 4351, 2914, 1176, 6967, 8025, 3764, 2493, 4020, 8790, 8091, 9677, 4647, 1285, 1979, 3627, 3383, 9356, 1917, 6091, 483, 8722, 7521, 908, 6771, 7532, 726, 7365, 7022, 4838, 3225, 5759, 8013, 1556, 9995, 5428, 656, 3929, 5782, 3190, 1103, 1127, 8528, 147, 2740, 2264, 9217, 3703, 8056, 5358, 5682, 1698, 4810, 2438, 2454, 2732, 1479, 4594, 9278, 22, 4054, 9270, 9287, 9043, 7860, 6563, 1014, 1992, 6096, 659, 4760, 4063, 7255, 852, 5347, 5842, 2050, 9529, 4401, 8470, 302, 4317, 4782, 5909, 2269, 2249, 7103, 9713, 2715, 3751, 2485, 790, 7124, 3240, 3171, 9567, 36, 7348, 5772, 4124, 639, 7744, 929, 1539, 4162, 4933, 5834, 7556, 2233, 3998, 1266, 7681, 4595, 3153, 7720, 6941, 5779, 354, 8595, 356, 2088, 8997, 8261, 2428, 5071, 2883, 1892, 11, 2551, 1470, 1477, 1161, 5, 4520, 3849, 8967, 4425, 54 ...snip... 4710, 1977, 9530, 1656, 5377, 4815, 5177, 7643, 3824, 4819, 9860, 8195, 3379, 6425, 5285, 1868, 3716, 1589, 7339, 1034, 2935, 1029, 1789, 2714, 4644, 484, 8748, 1417, 1271, 4237, 1512, 1987, 8040, 61, 7734, 462, 6695, 7212, 5937, 5400, 7706, 3308, 1503, 8479, 651, 35, 9429, 320, 4891, 1366, 4141, 6225, 9644, 4695, 7046, 6531, 1069, 3025, 5965, 6258, 4422, 1552, 1579, 4943, 2279, 8775, 3343, 6224, 2861, 3916, 8926, 9318, 101, 6358, 696, 9023, 5635, 8858, 2253, 4861, 341, 242, 4047, 2600, 7923, 4896, 1097, 5375, 1578, 1970, 145, 9988, 7114, 7158, 5418, 1113, 5437, 6751, 6509, 5605, 9929, 488, 2888, 1355, 6087, 2009, 3208, 717, 2788, 8344, 2212, 1085, 7016, 3857, 1097, 2389, 2662, 1321, 7533, 2200, 6579, 447, 7326, 2679, 91, 8709, 4411, 696, 640, 1102, 694, 1941, 9924, 3512, 3496, 1617, 9705, 5707, 5349, 2816, 1910, 6658, 1922, 6567, 1453, 2490, 8180, 1729, 8555, 1913, 6874, 8100, 9904, 8994, 6862, 1700, 5600, 7653, 5176, 7574, 3369, 2507, 9908, 328, 6373, 595, 8885, 3263, 9042, 3079, 9514, 1872, 465, 2995, 3695, 7027, 8076, 8294, 3191, 1285, 7963, 6713, 3801, 1693, 2201, 2784, 8649, 8321, 4626, 9231, 8488, 378, 4223, 128, 4238, 2534, 1542, 370, 3047, 4828, 76, 1619, 3100, 7269, 188, 5873, 9540, 5705, 6429, 6365, 2874, 4551, 3054, 8481, 8225, 2506, 2005, 1636, 3125, 5576, 2425, 4171, 9284, 1420, 1692, 9458, 1549, 4855, 1965, 5260, 743, 2370, 3901, 6231, 5125, 6759, 1553, 3764, 2856, 9521, 8407, 902, 1750, 3393, 6428, 8232, 1450, 7586, 5741, 2771, 9580, 399, 7453, 6992, 4963, 2228, 3847, 8939, 3016, 6419, 4612, 410, 8124, 2856, 6734, 6461, 9612, 8158, 9302, 4469, 6740, 1843, 6563, 6281, 418, 2505, 493, 6311, 8941, 9359, 6649, 3592, 2310, 4256, 9669, 471, 7085, 6314, 9415, 6115, 5895, 4722, 3489, 3059, 2734, 5238, 9911, 9258, 1130, 4882, 1380, 4044, 6751, 791, 7498, 1519, 1718, 9417, 8088, 5781, 8625, 536, 1393, 601, 1689, 6978, 539, 6711, 6906, 870, 3941, 1474, 5218, 92, 1340, 3168, 7350, 451, 7120, 6216, 8629, 3179, 7235, 4032, 4305, 933, 4002, 3040, 5643, 1720, 6326, 9035, 5358, 277, 8989, 7710, 1407, 3997, 3029, 7769, 9843, 2305, 6393, 7375, 9330, 153, 9647, 4489, 8906, 3113, 9936, 6335, 7767, 9362, 586, 8073, 1586, 517, 5683, 9139, 4424, 2606, 3823, 3952, 924, 5886, 5268, 9008, 2927, 8040, 540, 2261, 8239, 6241, 5322}, 7671, 7671, ✓				

Test	Expected	Got	
<pre>818, 7118, 898, 8625, 1797, 4037, 4287, 9521, 3255, 5816, 877, 8355, 9761, 6635, 2544, 39, 8289, 3605, 3634, 5163, 6669, 8500, 7723, 2255, 455, 3794, 205, 9048, 9826, 4135, 287, 4402, 6212, 782, 3917, 7225, 893, 764, 2558, 7689, 8023, 590, 6934, 8964, 2013, 9955, 6460, 6916, 3601, 2043, 894, 3666, 2649, 9261, 7060, 8791, 5418, 9327, 1651, 4960, 1128, 6497, 5732, 2423, 8141, 1371, 868, 8672, 3064, 9577, 4652, 8609, 6327, 451, 3411, 2342, 472, 6210, 5395, 6819, 2828, 4672, 1609, 6924, 77, 240, 1617, 1601, 1906, 3694, 1906, 6190, 5089, 8455, 3078, 530, 292, 9404, 9005, 2158, 1695, 5123, 2050, 8801, 8496, 7071, 718, 9921, 2872, 5297, 7227, 4300, 1129, 8671, 3067, 1264, 6465, 6467, 1099, 2536, 5334, 8839, 5026, 4920, 1650, 5601, 2047, 523, 3762, 9785, 2882, 8379, 2889, 4514, 9912, 2672, 5034, 2664, 4861, 9100, 4133, 2967, 3518, 103, 1073, 5597, 2621, 8983, 9276, 1601, 1656, 5795, 8527, 7251, 8971, 3767, 86, 7685, 1573, 6554, 2590, 9659, 953, 3998, 6543, 3487, 919, 6984, 8316, 8296, 4682, 3249, 4040, 204, 578, 4556, 4332, 9252, 1525, 3881, 5303, 6223, 1425, 3465, 6181, 7497, 8998, 7923, 6851, 8621, 9810, 2358, 7153, 2249, 3436, 9952, 808, 241, 3125, 3615, 8127, 7271, 4620, 8310, 2329, 9385, 8401, 338, 8289, 9515, 28, 1262, 2361, 8919, 4991, 9987, 399, 8690, 3193, 7476, 7729, 4784, 1356, 2329, 770, 9627, 4450, 5914, 7629, 2672, 4876, 5102, 1658, 1739, 1946, 5124, 2527, 8117, 6194, 8019, 9433, 9578, 2282, 5165, 3755, 1360, 3384, 2153, 693, 484, 8013, 5089, 2297, 9479, 4493, 3209, 5866, 3338, 5411, 7522, 7206, 69, 5426, 8029, 7974, 1630, 6766, 1571, 4723, 2632, 9612, 517, 1659, 5946, 5272, 8368, 9348, 450, 9622, 7411, 3985, 1385, 7211, 6628, 2850, 4633, 7788, 6863, 577, 9308, 3335}; cout << sumOfMinSubarray(nums);</pre>			
✓ <pre>vector<int> nums {7456, 3530, 1415, 8785, 9636, 6314, 5719, 2015, 9803, 4492, 1627, 3529, 8641, 6272, 130, 6044, 2381, 7261, 2790, 400, 2793, 6296, 2137, 137, 9459, 8246, 2942, 721, 275, 6606, 5708, 8070, 1598, 1618, 2622, 7103, 2282, 559, 559, 3461, 2404, 8435, 724, 7568, 5550, 1983, 5258, 4480, 2989, 3870, 3602, 4397, 3218, 9365, 2330, 7281, 6919, 5594, 2959, 6452, 12, 1369, 9551, 9528, 4562, 7509, 9299, 2822, 3449, 9229, 477, 5240, 3461, 4176, 6942, 2265, 9526, 3841, 9467, 6058, 1944, 2110, 3345, 9511, 5959, 1261, 2102, 8121, 9383, 3032, 8466, 8429, 8836, 6924, 5631, 8757, 8005, 2732, 8445, 2061, 199, 8251, 1417, 6016, 4103, 3130, 2526, 140, 3967, 8142, 6972, 1860, 9733, 405, 3864, 85, 6176, 3003, 2133, 6043, 429, 5256, 1874, 1012, 3553, 7197, 7587, 4101, 6784, 2607, 2292, 4620, 253, 8194, 1563, 5590, 5890, 7737, 3886, 7393, 7394, 4749, 2533, 261, 8350, 3741, 6705, 6911, 8613, 7477, 1949, 7262, 2367, 9682, 3588, 5009, 8860, 6292, 1299, 2272, 7256, 1781, 4465, 5051, 6450, 7033, 3484, 6592, 7422, 5926, 1589, 4977, 679, 3703, 4223, 8004, 1289, 2587, 3107, 5533, 8670, 7747, 552, 5289, 7319, 7516, 9874, 8748, 7341, 7418, 1496, 4510, 7288, 2086, 5514, 7122, 8448, 153, 1987, 1719, 498, 2185, 5899, 290, 2582, 2451, 657, 4667, 8015, 3978, 5703, 2407, 1330, 5982, 4093, 4774, 5067, 3761, 2227, 5934, 8396, 9494, 8985, 3192, 3478, 7451, 5690, 5513, 6557, 2041, 2424, 6499, 1192, 9257, 6777, 9949, 4215, 313, 6531, 9889, 2264, 432, 6820, 8741, 6698, 4526, 965, 7660, 2435, 1671, 6303, 3325, 7616, 545, 4047, 6292, 8396, 4524, 2166, 9656, 6211, 2898, 4532, 2061, 4310, 911, 8342, 4250, 1809, 6787, 1549, 1678, 8203, 5493, 1250, 2191, 5535, 5898, 2871, 1491, 2217, 8295, 4687, 3058, 1874, 6003, 1344, 6316, 8880, 6512, 6280, 5431, 7960, 2028, 7518, 7899, 6165, 1938, 1643, 5474, 9485, 8775, 1496, 874, 2578, 2034, 2409, 4093, 8182, 1723, 7595, 9411, 8707, 3400, 6890, 4790, 7002, 8707, 3318, 6902, 507, 4880, 1831, 702, 1940, 3374, 2206, 8039, 6269, 923, 1886, 5115, 8179, 9517, 8235, 5613, 2563, 2879, 6178, 3303, 7338, 8264, 697, 4628, 4939, 8850, 4888, 1855, 112, 6138, 185, 1756, 356, 2115, 6779, 4245, 8945, 1056, 1661, 2104, 8406, 1938, 2281, 3887, 9483, 734, 1407, 5167, 9687, 7411, 2596, 9590, 6354, 7498, 3818, 3182, 8215, 9231, 9282, 9575, 4508, 415, 3775, 7412, 8709, 4498, 6013, 9589, 9349, 9371, 562, 8085, 631, 5614, 1891, 6181, 2877, 482, 3963, 6104, 8358, 6039, 4167, 1936, 933, 7737, 9141, 2249, 6272, 5685, 367, 4181, 310, 3835, 930, 6008, 116, 6565, 1348, 6708, 4186, 6183, 1365, 4040, 8665, 8072, 2245, 1140, 4656, 6992, 9399, 3361, 3272, 194, 2020, 4078, 9132, 7468, 9395, 766, 3885, 4844, 3671, 2192, 2757, 7797, 1801, 2511, 3402, 1107, 1528, 4796, 1302, 5067, 9839, 856, 804, 1991, 1798, 8822, 9005, 3137, 4523, 1936, 2005, 6797, 5282, 665, 4127, 9819, 618, 1571, 1800, 9901, 579, 8274, 4978, 9414, 9853, 4721, 5055, 244, 4994, 9491, 1970, 1472, 2295, 2253, 8544, 8308, 3166, 6802, 7996, 1964, 9456, 5198, 8180, 4863, 1705, 8328, 9750, 5327, 2164, 2033, 7799, 2304, 7706, 1842, 7989, 9366, 1861, 502, 4139, 6050, 7113, 5651, 4277, 1022, 8357, 319, 9349, 8982, 1430, 800, 6053, 2858, 4861, 1044, 5230, 593, 5970, 146, 3881, 1936, 3929, 3425, 7331, 8988, 292, 5992, 6854, 6812, 2377, 1622, 150, 2690, 4645, 7347, 6969, 8444, 961, 1670, 491, 1462, 3061, 4056, 7184, 3682, 7482, 8504, 9787, 5051, 7988, 1185, 591, 3925, 9119, 8777, 2884, 8985, 4345, 3036, 61, 7249, 4801, 1609, 6855, 4287, 2351, 7866, 1140, 1410, 820, 6826, 9673, 8059, 6995, 6258, 1589, 5869, 9072, 6676, 9987, 9252, 6295, 188, 3537, 6823, 4256, 2259, 8450, 4049, 8962, 38, 1217, 9116, 7542, 2359, 6759, 7055, 846, 6946, 239, 9465, 261, 835, 8872, 5502, 6328, 8006, 747, 6563, 230, 254, 9934, 772, 1892, 650, 5959, 9726, 6404, 870, 1221, 2489, 5702, 4974, 3515, 9958, 7855, 8600, 1102, 7162, 3210, 3538, 5602, 6793, 3853, 9881, 8800, 2615, 1085, 188, 2331, 2011, 2100, 6768, 8003, 2677, 9938, 9749, 4521, 1905, 5937, 1673, 5428, 2572, 8590, 7804, 5267, 262, 8005, 660, 2628, 2873, 2220, 5810, 917, ...snip..., 4943, 6829, 4804, 2446, 593, 2161, 5353, 9464, 1396, 3831, 8538, 189, 3166, 5550, 6153, 1302, 3287, 8203, 6197, 9377, 1152, 6479, 9009, 2124, 5556, 8630, 8451, 8914, 2334, 4272, 250, 6021, 1003, 5158, 9364, 766, 3249, 149, 6318, 6145, 2288, 5967, 7613, 9700, 1208, 5811, 5229, 4177, 1020, 7785, 9256, 969, 8934, 4434, 5738, 1756, 1338, 9775, 530, 9319, 5215, 9153, 827, 6621, 8891, 4289, 9883, 2713, 62, 6706, 3996, 9906, 5827, 9571, 4505, 9536, 3554, 9641, 2635, 4942, 7513, 7005, 1174, 9889, 4093, 2626, 1885, 4440, 8491, 9241, 4813, 2677, 7763,</pre>	9382	9382	✓

Test	Expected	Got	
<pre> 8376, 5129, 3019, 1224, 104, 7537, 1255, 321, 6622, 7736, 2580, 9172, 1981, 1558, 4250, 7507, 5466, 4060, 4727, 3735, 80, 7494, 465, 3533, 6837, 2133, 7098, 2750, 3549, 2702, 4052, 2979, 4973, 5021, 583, 5625, 5154, 9470, 386, 3009, 8371, 7549, 638, 1099, 8900, 924, 2185, 8996, 8157, 309, 2066, 9501, 6827, 9729, 816, 5550, 997, 745, 4055, 7627, 9281, 3259, 2301, 8022, 8424, 7487, 3786, 5798, 6519, 4329, 5119, 6806, 9640, 3026, 729, 928, 6096, 2419, 2710, 8108, 1802, 1777, 5671, 2403, 7124, 895, 7318, 3673, 6552, 5408, 5978, 2715, 4159, 8125, 1768, 1445, 7425, 1862, 7276, 5991, 9647, 2268, 8729, 4553, 7332, 1646, 8605, 220, 2169, 1202, 680, 163, 2473, 6743, 8895, 6114, 6453, 2744, 5676, 2822, 8632, 3395, 2008, 917, 3197, 2139, 3356, 4263, 9888, 2604, 3108, 3765, 7282, 3869, 5510, 8785, 337, 3242, 2961, 7686, 4087, 4562, 280, 1873, 7154, 1374, 6961, 9313, 3124, 8532, 4314, 7611, 1580, 7099, 2092, 145, 7013, 1647, 1762, 4420, 173, 4823, 5211, 7223, 536, 2293, 6318, 662, 8454, 2766, 1598, 1511, 8883, 2363, 8414, 2415, 7581, 1423, 5270, 6760, 1051, 250, 2679, 9586, 8956, 4029, 188, 7160, 1822, 1727, 1633, 6788, 4624, 448, 7688, 7782, 6122, 1132, 1962, 2112, 6235, 7522, 8027, 8726, 3651, 437, 5777, 9588, 1938, 530, 3862, 2064, 1441, 8627, 9548, 9986, 3536, 6820, 8288, 6497, 1135, 8965, 6946, 3168, 3265, 9295, 6516, 168, 8046, 8250, 8114, 3688, 7448, 8237, 9329, 9675, 6624, 3351, 6858, 6425, 394, 4687, 184, 8304, 6638, 5895, 3648, 8676, 7312, 3223, 6865, 8772, 9602, 5911, 9933, 2287, 872, 6299, 7847, 6887, 8685, 7478, 397, 2869, 6773, 9689, 5527, 2478, 3330, 7179, 7168, 2885, 4027, 5005, 789, 176, 6281, 254, 3036, 3519, 3188, 6109, 8263, 5146, 9166, 6621, 6617, 2570, 3134, 1518, 2472, 9291, 480, 6921, 8227, 6832, 3642, 76, 4054, 5800, 7666, 1191, 4940, 320, 5902, 486, 708, 153, 8380, 6944, 1255, 3771, 1588, 7935, 6697, 89, 7371, 6657, 1278, 1063, 7721, 7358, 288, 6506, 4237, 651, 2046, 2448, 3935, 1647, 4214, 9920, 8054, 4468, 501, 9778, 1068, 9712, 1310, 6785, 1006, 1636, 5642, 8384, 9384, 4581, 2096, 5852, 7361, 313, 9222, 2816, 4291, 4116, 9670, 8206, 5294, 4244, 8851, 5213, 2748, 5610, 5486, 126, 9592, 9290, 9668, 2004, 1019, 1924, 8603, 6402, 5064, 7044, 4557, 7703, 9897, 8743, 723, 760, 2344, 3391, 6774, 4269, 6178, 1672, 5330, 2911, 121, 4780, 8365, 5855, 4394, 4452, 9222, 2251, 6367, 9441, 9281, 4669, 933, 3178, 1538, 8644, 9824, 1846, 7383, 4535, 125, 3772, 6251, 9223, 1128, 4889, 1289, 8788, 3168, 3330, 9390, 4147, 792, 2539, 6866, 2031, 9631, 4087, 1416, 8206, 9138, 4672, 4435, 6837, 736, 2953, 939, 9230, 1855, 3459, 1761, 1384, 125, 2605, 5296, 5120, 5353, 8080, 1140, 2207, 2563, 9209, 4144, 4292, 1138, 2072, 6122, 20, 8436, 2777, 1001, 447, 3195, 4544, 6786, 2804, 4038, 315, 7807, 1656, 6093, 3571, 6354, 9379, 9794, 3299, 3577, 9483, 9492, 507, 5730, 8982, 2581, 4551, 6528, 729, 8894, 2204, 1694, 6870, 7234, 8371, 8788, 7510, 2193, 6282, 3480, 1027, 6430, 9446, 9478, 2730, 7538, 2605, 9155, 4879, 2082, 4564, 6990, 5585, 847, 7380, 5481, 8799, 7885, 2482, 2583, 6923, 4630, 2191, 5702, 5486, 7199, 2141, 7454, 2940, 9852, 1379, 986, 1354, 2720, 9651, 8999, 2573, 6791, 4186, 5339, 8404, 3623, 4696, 537, 2777, 5091, 6481, 1841, 5508, 659, 1251, 4061, 9649, 6209, 2793, 217, 1235, 7653, 4943, 2626, 1896, 2723, 8374, 5731, 1631, 8627, 2586, 1291, 1848, 4514, 8979, 2179, 7403, 7495, 4345, 2159, 3723, 2425, 8795, 52, 7116, 1231, 1458, 1817, 3459, 9863}; cout << sumOfMinSubarray(nums); </pre>			
✓ <pre> vector<int> nums {4577, 9408, 3645, 8203, 9984, 1798, 9011, 8496, 698, 3465, 8332, 161, 563, 5914, 1763, 2509, 6382, 2435, 3483, 3228, 7581, 4443, 6096, 4183, 4042, 8071, 1328, 7449, 8822, 9651, 4522, 1888, 3009, 7394, 4709, 6600, 1129, 2642, 3300, 6619, 4358, 5428, 7624, 1473, 5477, 2603, 2375, 4334, 8995, 2658, 3484, 7869, 2526, 4610, 1951, 1767, 3446, 3188, 5543, 841, 4185, 4933, 8476, 6948, 3359, 392, 5042, 8118, 4595, 1936, 8787, 7674, 1331, 9421, 2604, 8699, 7114, 5734, 1515, 7455, 9228, 425, 5639, 9378, 722, 4527, 48, 2176, 6967, 947, 1321, 2151, 3505, 6276, 2579, 9771, 9636, 4569, 492, 6172, 1715, 3861, 5576, 5078, 9561, 5964, 8830, 9960, 4840, 8788, 4791, 6631, 9920, 6031, 5106, 4931, 6065, 3464, 9068, 2473, 8836, 5148, 9918, 8571, 7863, 5867, 1863, 7918, 9048, 5422, 1067, 1698, 644, 8316, 1548, 5079, 8496, 7672, 7721, 4286, 1448, 957, 9428, 4989, 950, 4637, 3976, 920, 2037, 8101, 4394, 5324, 6486, 6149, 3983, 9277, 691, 1600, 4671, 4547, 5842, 3935, 3704, 2080, 1261, 5994, 3430, 3989, 1856, 7113, 8690, 8230, 1010, 3304, 7657, 2955, 5399, 1354, 6798, 2953, 5, 9759, 3026, 6224, 8853, 7033, 1395, 9893, 541, 4939, 2659, 2883, 236, 419, 2266, 5614, 9189, 6074, 2959, 6489, 3792, 6185, 2547, 9256, 9325, 1534, 7790, 2088, 619, 7855, 6750, 5854, 1992, 4398, 9741, 9037, 6377, 206, 8658, 5260, 1084, 8597, 8027, 4519, 3948, 2371, 314, 884, 9013, 4066, 7949, 89, 1033, 2270, 9795, 4045, 707, 1526, 5915, 8476, 2697, 4067, 5007, 895, 4071, 8153, 2625, 9387, 1002, 3565, 7509, 6664, 3881, 8854, 6288, 1109, 7435, 2150, 2799, 154, 2133, 1194, 9530, 2367, 7971, 3776, 7866, 4027, 8426, 5497, 6969, 6575, 6459, 943, 8602, 8564, 8506, 3497, 313, 9570, 7871, 8797, 1075, 7379, 2845, 2165, 6265, 7143, 9889, 2069, 7137, 6601, 5069, 7787, 6606, 3729, 3270, 5672, 8353, 1548, 7035, 4142, 895, 9209, 5463, 459, 9797, 5797, 8841, 1381, 282, 6410, 6401, 8614, 4948, 7123, 4237, 5163, 1790, 2707, 9238, 6535, 440, 3687, 7765, 8813, 4159, 5721, 7098, 8413, 5311, 934, 4510, 4494, 3718, 962, 7542, 1150, 5018, 1515, 7579, 1862, 2254, 2331, 7512, 3986, 8881, 2066, 2212, 3583, 1080, 723, 7681, 7141, 5677, 5789, 1444, 2183, 7719, 4413, 5590, 6248, 4188, 5731, 1178, 6529, 7541, 322, 3728, 1505, 7617, 3852, 8505, 9636, 7564, 5357, 5324, 6903, 4017, 358, 3273, 570, 4590, 9710, 3388, 342, 6583, 6500, 755, 2146, 6125, 4247, 9156, 8913, 2897, 8697, 2489, 9326, 6708, 306, 9325, 2107, 7872, 4625, 1596, 9574, 2556, 3576, 997, 5917, 2268, 6601, 9043, 9443, 8856, 2715, 6029, 2548, 5518, 4162, 5467, 1429, 8762, 2494, 4170, 5471, 3509, 3506, 6241, 2142, 4411, 600, 8373, 3493, 5761, 4867, 3642, 8420, 2423, 3445, 1659, 2301, 8635, 4610, 9035, 5395, 958, 9460, 4013, 9492, 4539, 5265, 2161, 9097, 1093, 6490, 5745, 475, 7792, 1349, 891, 1640, 7155, 7117, 3647, 8294, 5420, 7592, 2091, 328, 4596, 8714, 405, 6655, 6117, 7417, 5113, 1026, 4057, 1424, 4484, 9632, 2629, 3815, 3616, 8853, </pre>	5989	5989	✓

Test	Expected	Got
822, 1939, 6032, 2268, 8906, 9866, 1711, 9125, 9219, 2442, 9428, 1863, 679, 2266, 1209, 9623, 8123, 1505, 7372, 2298, 9106, 1734, 1493, 6343, 6320, 1223, 6089, 752, 8468, 4734, 6655, 114, 2806, 7377, 4849, 7451, 2476, 377, 1861, 5004, 963, 9221, 8732, 6112, 2300, 7568, 8682, 9740, 2597, 4000, 8664, 9110, 8922, 1793, 7511, 3227, 7532, 1537, 6214, 3355, 9361, 461, 1525, 8632, 1371, 1404, 4586, 880, 1053, 8954, 5156, 5398, 3292, 7400, 543, 8440, 5203, 8626, 1784, 4284, 2617, 3985, 9121, 7996, 6582, 9970, 3158, 965, 2484, 7878, 977, 7274, 611, 1579, 2523, 334, 9695, 8220, 8068, 9506, 1350, 5349, 9461, 1067, 7653, 825, 884, 40, 4002, 2551, 137, 1506, 7167, 4842, 7854, 5348, 29, 3393, 8011, 9022, 6512, 4126, 771, 7829, 1682, 6194, 137, 8557, 8730, 4857, 2391, 7465, 2542, 8530, 3297, 9106, 8806, 642, 8981, 3127, 243, 7369, 6789, 4292, 7472, 4485, 4409, 4118, 1368, 5638, 6766, 5965, 6568, 8690, 9188, 1363, 3430, 3321, 4439, 8508, 8275, 1179, 619, 990, 7965, 9449, 302, 5092, 4824, 8058, 5560, 3709, 2397, 8623, 1606, 5706, 835, 7497, 904, 5513, 5166, 2265, 9239, 3307, 4613, 3880, 4091, 5244, 9176, 3960, 78 ...snip... 8, 3308, 1401, 3288, 8261, 4429, 9001, 1116, 1266, 5139, 3814, 8589, 2757, 1233, 8337, 6007, 9761, 8997, 7845, 4942, 9622, 5063, 4053, 9636, 5053, 1362, 2901, 7760, 5734, 4404, 7357, 6092, 2815, 4801, 3609, 2189, 9481, 5836, 2643, 4841, 7750, 8664, 21, 7123, 6565, 635, 3706, 2106, 81, 6660, 1525, 548, 1206, 543, 4884, 8485, 1014, 5751, 5238, 52, 139, 5324, 4394, 1280, 1262, 1909, 3642, 1902, 5848, 2263, 7512, 7543, 3622, 1046, 104, 7600, 2146, 2540, 7022, 9065, 6928, 5288, 1929, 3290, 5606, 8296, 1825, 1914, 1728, 1466, 685, 675, 403, 1663, 3880, 6851, 912, 8353, 1300, 7380, 1323, 8951, 6703, 446, 1260, 8438, 7190, 3290, 6181, 6171, 8807, 4138, 7798, 5494, 6475, 1880, 4878, 1417, 3780, 3101, 323, 6242, 8068, 7507, 3844, 5812, 8526, 7325, 8092, 6109, 6080, 1093, 7753, 1315, 465, 1446, 5614, 6320, 1335, 3915, 2166, 2624, 1197, 1879, 5083, 617, 9507, 7015, 662, 7643, 1270, 821, 6967, 4999, 1442, 2110, 5670, 340, 1589, 5590, 635, 5408, 7173, 9194, 2108, 4948, 5749, 4663, 2069, 5659, 7828, 8409, 4253, 405, 6870, 743, 3750, 551, 6760, 3069, 2485, 9349, 4286, 1145, 563, 4754, 1791, 794, 1747, 2455, 7939, 7366, 6805, 3043, 7931, 7052, 938, 1778, 6659, 889, 4745, 1216, 580, 2158, 4482, 8192, 9052, 3713, 1968, 8984, 1970, 6403, 9460, 9087, 4436, 8515, 1520, 8011, 1491, 8017, 9217, 1236, 9154, 1205, 4674, 6829, 8635, 2017, 1452, 3908, 9539, 980, 7570, 8775, 3199, 6993, 8940, 1001, 1511, 2143, 6642, 5093, 4729, 9826, 1355, 1283, 8337, 202, 3446, 1065, 5383, 8522, 4186, 5940, 3962, 2956, 8281, 71, 6492, 2355, 5978, 3618, 4046, 8625, 4903, 9607, 8990, 2266, 9908, 6793, 4093, 4680, 8946, 9078, 9862, 5360, 290, 4999, 1452, 6310, 6989, 8469, 9699, 5197, 9572, 3230, 561, 2369, 7636, 9864, 8406, 1080, 3660, 5061, 5915, 7574, 5378, 4312, 5415, 8240, 6482, 8168, 7192, 1108, 3256, 2162, 3584, 1479, 3505, 2243, 9789, 9775, 5997, 8566, 5901, 1218, 648, 1643, 1537, 6493, 9785, 7097, 5442, 426, 4298, 7029, 8658, 9948, 3593, 6211, 6869, 1629, 2878, 5133, 4511, 4043, 990, 4898, 7343, 1406, 591, 4128, 7132, 6365, 8590, 7957, 6847, 2369, 6109, 9890, 1714, 5209, 2209, 8146, 1127, 3536, 5579, 5721, 4636, 7932, 8421, 5184, 205, 6664, 6735, 9568, 299, 629, 9530, 5021, 8443, 6508, 6262, 9174, 578, 216, 1623, 4265, 2320, 258, 1469, 8556, 1994, 6764, 4964, 5946, 7613, 8305, 2008, 4746, 8448, 2076, 1933, 2134, 905, 5721, 6975, 3531, 9346, 1914, 4998, 2364, 2189, 5675, 5884, 6076, 8838, 9992, 7921, 8907, 1801, 1869, 126, 5633, 9363, 6701, 3001, 8921, 2285, 1847, 8745, 5120, 6310, 230, 3243, 6222, 6912, 8696, 8708, 947, 2527, 6242, 5016, 2246, 2509, 4242, 8874, 2634, 7329, 8261, 7348, 1555, 1696, 6170, 3369, 3752, 3045, 8934, 1082, 2056, 6686, 4953, 534, 6962, 2215, 5841, 197, 4863, 5867, 6398, 3863, 5923, 8487, 325, 3176, 1012, 5793, 1225, 839, 9594, 5849, 2119, 200, 2904, 8112, 1345, 9302, 5721, 3878, 4670, 2701, 5823, 9288, 399, 5543, 568, 698, 7957, 1077, 4802, 6800, 1617, 2256, 7201, 4724, 5404, 112, 6030, 8131, 392, 1437, 9570, 1976, 7030, 9348, 7443, 5774, 330, 3917, 6222, 480, 9020, 7323, 4486, 2723, 177, 3219, 6411, 5143, 5689, 6737, 8271, 8441, 9130, 1591, 5397, 6509, 6621, 2423, 1227, 1466, 1959, 6514, 7050, 3830, 1291, 2562, 5147, 4644, 1437, 3862, 7340, 4778, 8858, 3148, 3331, 4856, 9959, 1972, 1299, 1174, 6519, 8465, 5757, 9214, 1857, 1923, 4072, 9658, 6417, 2676, 3252, 1279, 9822, 3200, 2235, 5610, 6028, 4648, 5321, 8841, 499, 5368, 5806, 2013, 2191, 5415, 8673, 2174, 4633, 4256, 7751, 3264, 5222, 1823, 7076, 8933, 5775, 125, 5890, 3300, 2833, 4944, 5312, 3368, 2629, 4813, 7196, 2060, 2435, 1919, 9988, 2181, 9794, 4252, 8522, 5200, 9859, 9798, 5078, 3630, 6607, 3184, 7392, 1964, 2174, 1105, 1998, 9880, 9466, 2030, 6458, 2642, 3971, 7571, 8872, 1535, 2708, 4257, 3110, 5700, 2838, 9423, 4965, 9873, 8052, 2522, 1496, 6203, 9998, 1775, 3675, 2334, 3666, 4219, 2021, 7299, 6119, 2113, 2384, 1263, 1804, 7515, 7721, 668, 9073, 4868, 6419, 6508, 7140, 6790, 6865, 667, 4095, 8176, 9860, 2356, 8751, 9002, 3681, 4662, 1880}; cout << sumOfMinSubarray(nums);		
✓ vector<int> nums {4491, 1079, 1097, 1112, 7947, 2543, 3676, 3098, 1077, 2821, 3680, 8107, 4351, 102, 6443, 1906, 2653, 4234, 406, 6683, 1090, 7220, 3335, 6965, 3465, 9768, 8694, 6285, 4868, 3133, 762, 4842, 898, 4930, 2420, 7930, 70, 8342, 3279, 4749, 6806, 1682, 2965, 4609, 5425, 7594, 4038, 4605, 2708, 4085, 5045, 8547, 7119, 344, 8931, 3967, 3779, 3577, 3184, 4261, 7487, 851, 3534, 8282, 2684, 9430, 4665, 2321, 8645, 2369, 8836, 2777, 1307, 1138, 9110, 2293, 8744, 7132, 6373, 2140, 8323, 1531, 1937, 1922, 3795, 1508, 3220, 8759, 7415, 3969, 1513, 1038, 5508, 6939, 7019, 2737, 1969, 4264, 6959, 7235, 9616, 7752, 6388, 22, 5326, 8843, 3497, 4126, 7437, 293, 6912, 9542, 4287, 8692, 9523, 5218, 4087, 1697, 2516, 7828, 3098, 1877, 9065, 8658, 1004, 4447, 1335, 6040, 3448, 147, 7054, 3770, 1279, 7796, 5286, 9389, 3877, 6245, 2744, 3990, 6984, 8829, 2432, 2979, 4131, 7641, 402, 521, 1618, 3767, 6442, 7726, 1063, 3728, 8152, 8376, 5628, 5524, 7148, 1621, 2250, 4349, 5167, 7772, 9818, 4909, 5426, 1076, 5177, 8585, 7626, 4980, 7985, 8059, 252, 5915, 118, 2546, 4102, 7441, 2366, 5213, 7935, 605, 8581, 808, 8015, 5128, 1695, 5269, 8434, 5746, 5007, 6862, 2904, 4307, 1000, 1674},	8789	8789 ✓

Test	Expected	Got
17, 3653, 1714, 6591, 3080, 4944, 6160, 4552, 5469, 4646, 6015, 7526, 2615, 3729, 7425, 939, 474, 1449, 587, 8821, 4821, 4138, 1100, 5112, 2479, 7509, 2623, 4239, 7108, 2118, 1250, 5316, 8346, 4054, 4712, 2014, 9273, 5410, 5116, 8443, 7562, 1068, 1559, 6030, 6558, 9120, 3012, 490, 851, 3610, 4594, 5568, 1667, 6006, 858, 4965, 7117, 1608, 6434, 4056, 472, 9165, 6576, 467, 1941, 6513, 6899, 5935, 4997, 4354, 8046, 2813, 2715, 2677, 758, 9762, 1779, 5336, 7695, 818, 3634, 2200, 5778, 9873, 1344, 8852, 3106, 8309, 724, 5569, 351, 2144, 3694, 8160, 8106, 6398, 1179, 8093, 4202, 9385, 1508, 4504, 3262, 5381, 4264, 7033, 6356, 8681, 1052, 544, 5162, 3514, 5008, 9003, 6355, 511, 804, 4476, 7868, 1568, 4986, 9754, 5951, 8128, 1629, 5495, 7624, 4253, 9747, 1457, 2358, 2725, 4223, 2269, 2500, 8508, 7786, 4529, 9833, 2075, 5779, 6592, 9757, 347, 6164, 332, 4509, 3890, 6695, 6663, 7520, 8573, 7673, 4586, 1850, 7119, 526, 1242, 3729, 889, 1877, 4871, 7535, 1803, 8253, 503, 6008, 1961, 2168, 3923, 8636, 6417, 5359, 2848, 2639, 9420, 1606, 865, 5463, 836, 9826, 2875, 6789, 5029, 4328, 1625, 2632, 3558, 1864, 5761, 5679, 2266, 9577, 3894, 173, 3490, 102, 6815, 5761, 7381, 2378, 1272, 4487, 3043, 7817, 6824, 5625, 4550, 6383, 2579, 2106, 4267, 7932, 2561, 8293, 1279, 1040, 9727, 4385, 4605, 1283, 7556, 2312, 6215, 6472, 5222, 2216, 1219, 4275, 2494, 7785, 9352, 1161, 1001, 608, 8776, 3639, 3552, 115, 1425, 1622, 8587, 7134, 5359, 1213, 4386, 6998, 7844, 5512, 580, 1932, 1211, 5815, 4347, 357, 9049, 1194, 79, 8807, 3909, 3899, 810, 3222, 1561, 108, 7596, 3424, 5929, 243, 1745, 5397, 3140, 8995, 9296, 9211, 3545, 1262, 3460, 5761, 9884, 4920, 7008, 1603, 888, 493, 1374, 1661, 647, 2917, 3593, 9363, 4272, 1799, 1485, 3873, 641, 5819, 8379, 6398, 3374, 6437, 2097, 3111, 4664, 719, 3481, 5737, 9280, 2134, 1345, 9790, 7198, 5288, 3595, 5692, 3870, 1740, 2022, 3194, 3834, 2435, 8188, 2036, 6046, 4957, 685, 1121, 2327, 975, 1805, 8964, 3773, 6260, 3449, 2784, 1140, 7857, 9984, 5615, 9581, 9277, 4351, 9413, 6994, 8721, 1597, 768, 4178, 140, 1657, 1506, 6074, 11, 6353, 6482, 8453, 2823, 6564, 4240, 2532, 6031, 8089, 6520, 5281, 7912, 466, 9065, 5120, 945, 476, 6487, 4184, 604, 9704, 7309, 1121, 2330, 4030, 9366, 9631, 1510, 4864, 6574, 9470, 293, 8942, 8325, 45, 9004, 3721, 2084, 9039, 291, 8585, 8739, 9840, 985, 7650, 6186, 8262, 4904, 2228, 7495, 2152, 5418, 9379, 6815, 8835, 6539, 2279, 5528, 6298, 289, 5444, 2422, 3534, 2938, 2412, 7461, 2095, 4037, 3203, 4167, 2577, 836, 7476, 6797, 7651, 3072, 2563, 6267, 5164, 2429, 8114, 984, 8038, 8255, 9032, 851, 4031, 7853, 3326, 2028, 8510, 613, 3644, 6498, 9651, 913, 516, 6985, 5202, 921, 812, 2351, 210, 825, 1778, 7838, 9937, 387, 6443, 6548, 2298, 6760, 1069, 9105, 4649, 2276, 3534, 4689, 8014, 551, 4221, 3148, 7153, 9083, 8927, 5834, ...snip... 1671, 4741, 6, 2472, 6725, 4579, 5642, 3249, 407, 1047, 878, 8858, 95, 7199, 5685, 52, 6209, 4159, 2708, 8932, 5335, 9063, 6687, 1999, 7912, 115, 383, 1383, 9290, 2417, 9719, 4148, 4005, 6472, 9383, 4038, 3338, 7463, 4384, 7513, 4740, 1516, 2200, 7562, 1054, 4454, 9313, 6506, 886, 1389, 8454, 2801, 6352, 2032, 1445, 8324, 4029, 5791, 9491, 8202, 1486, 3051, 9180, 6892, 7363, 8394, 7770, 7051, 8473, 8960, 6457, 438, 4804, 3498, 6888, 3084, 3593, 2124, 7490, 8094, 5503, 6782, 2542, 4217, 8070, 6706, 1811, 9507, 5001, 777, 128, 1443, 1392, 5335, 5499, 7478, 3018, 8276, 2825, 3968, 2953, 2020, 2270, 9616, 9406, 2601, 2917, 1383, 166, 408, 3345, 6393, 1829, 8138, 6997, 6996, 6162, 3187, 9057, 8226, 7139, 8792, 8254, 346, 4395, 1308, 2445, 6188, 6624, 3948, 3706, 2732, 66, 7741, 8423, 8350, 5130, 3426, 5302, 8371, 9087, 8722, 9590, 387, 8334, 6089, 5347, 4284, 5995, 4637, 1271, 7337, 2835, 6388, 2405, 8566, 5572, 4996, 9975, 707, 6079, 1612, 8310, 3384, 7261, 4737, 27, 1816, 4640, 7243, 5028, 9289, 3899, 7394, 2361, 3699, 5109, 4098, 2623, 9174, 7097, 9591, 7196, 9873, 9489, 1253, 8780, 1856, 1397, 1334, 4868, 7830, 2591, 4762, 3269, 3148, 8913, 2538, 3924, 9832, 1145, 9854, 1815, 4865, 2307, 4272, 4610, 4593, 15, 4000, 9826, 6861, 8693, 1336, 5162, 5202, 6118, 2835, 3389, 6992, 805, 10, 9019, 59, 8982, 5264, 6719, 4111, 6326, 5423, 8232, 2388, 1164, 2657, 2423, 9618, 8942, 884, 7176, 8743, 9766, 4920, 6693, 2487, 6934, 9610, 4151, 8403, 3977, 4486, 2670, 3469, 6893, 9928, 9443, 3466, 4782, 7812, 8300, 5545, 7733, 1352, 4496, 1348, 8993, 9886, 1275, 1861, 8132, 5206, 6020, 6459, 6081, 1587, 9998, 524, 2132, 6132, 2732, 892, 5939, 6854, 104, 111, 9683, 3672, 3813, 5372, 6671, 4400, 851, 1588, 1061, 268, 1076, 5985, 2240, 4140, 2605, 2625, 5597, 7198, 2222, 5461, 1224, 4329, 9718, 6469, 7063, 6302, 4593, 9799, 9883, 6187, 5139, 3999, 8898, 3043, 3849, 3897, 6335, 923, 1486, 3557, 1166, 8565, 9755, 2979, 130, 8082, 6611, 5039, 9522, 8804, 1982, 878, 1844, 6306, 4564, 8538, 8664, 4736, 1309, 1353, 7001, 1566, 8726, 469, 5647, 4591, 7463, 2872, 3240, 5037, 8819, 2103, 7064, 9082, 9135, 881, 8591, 4442, 9462, 2183, 5018, 112, 1084, 6941, 1292, 743, 290, 1506, 798, 802, 6809, 9197, 7333, 2133, 555, 8648, 6276, 5781, 6614, 770, 1233, 1069, 4666, 1664, 4700, 6644, 8470, 1187, 7933, 4315, 4366, 928, 1357, 8731, 5967, 827, 4537, 6462, 3078, 3941, 4258, 2012, 142, 1609, 3160, 8170, 9048, 5921, 8520, 3294, 6688, 7934, 3635, 5109, 653, 6632, 5249, 2521, 9369, 1730, 6246, 1629, 498, 7474, 7508, 5947, 2657, 4352, 7089, 7889, 6153, 9901, 9531, 3398, 8242, 9813, 8377, 19, 1366, 6232, 2579, 8560, 2651, 4885, 5381, 1612, 6067, 3496, 5744, 9970, 5792, 6054, 2161, 7608, 5772, 742, 9366, 5033, 600, 1203, 7774, 6742, 8707, 7231, 1265, 904, 2093, 1370, 1888, 5322, 1709, 6244, 7278, 1156, 1824, 4015, 6415, 7866, 3, 1464, 5679, 5927, 9587, 4039, 391, 7380, 9602, 7908, 9755, 5710, 3293, 781, 3192, 8647, 1436, 4806, 3804, 5227, 1641, 4016, 5910, 5621, 1941, 1949, 6672, 947, 4951, 659, 5529, 6760, 4758, 9071, 7016, 6544, 7721, 1275, 3422, 3369, 7373, 5475, 5660, 4145, 4887, 3285, 6021, 986, 1003, 9752, 5653, 3509, 3645, 5662, 8738, 6990, 6581, 7596, 8977, 8708, 9592, 3721, 6524, 1244, 6900, 4805, 7571, 9958, 9516, 5229, 9825, 8064, 1509, 5436, 6528, 8489, 1888, 164, 2255, 9513, 8346, 2989, 5011, 9344, 9984, 557, 6444, 3868, 7494, 153, 9338, 4430, 1194, 3354, 4815, 2639, 8965, 6484, 390, 7983, 6642, 2467, 8709, 5087, 4866, 3575, 8735, 3344, 8686, 6433, 2893, 8347, 3135, 6462, 7006, 5538, 4891, 1428, 1180, 6294, 6832, 1451, 9826, 7323, 326, 3278, 8059, 4025, 2224, 178, 8219, 7189, 4196, 8038, 2500,		

Test	Expected	Got
8127, 107, 6699, 7282, 2525, 2027, 2821, 552, 6136, 3779, 371, 927, 4117, 3254, 4572, 5643, 8838, 2395, 6043, 2209, 5810, 2676, 156, 9990, 7923, 4991, 5751, 9658, 2286, 4090, 3325, 1397, 937, 330, 7104, 7936, 6296, 3773, 9160, 5623, 2664, 189, 1475, 1293, 9087, 2312, 9229, 3073, 7203, 5263, 9504, 2415, 4763, 3516, 5293, 426, 1379}; cout << sumOfMinSubarray(nums);		

Passed all tests! ✓

[Chỉnh xác](#)

Điểm cho bài nộp này: 1.00/1.00.

Câu hỏi 14

Chính xác

Điểm 1,00 của 1,00

Given an array of integers.

Your task is to implement a function with following prototype:

```
bool tripleIndexSLM(vector<int>& nums);
```

The function returns whether a triplet of indices (*i*, *j*, *k*) such that *i* < *j* < *k* and *nums*[*i*] < *nums*[*k*] < *nums*[*j*] exists.

Note:

- The `iostream`, `vector` and `stack` libraries have been included and `namespace std` is being used. No other libraries are allowed.
- You can write helper functions.

For example:

Test	Result
<code>vector<int> nums {1, 10, 0, 3, 3}; cout << tripleIndexSLM(nums);</code>	1

Answer: (penalty regime: 0 %)

Reset answer

```
1 ✓ bool tripleIndexSLM(vector<int>& nums) {
2     // STUDENT ANSWER
3     if(nums.size()<=2){
4         return 0;
5     }
6     vector<int> v;
7     v.resize(2);
8     deque<int> de;
9     if(nums[0]<=nums[1]){
10        v[0]=nums[0];
11        v[1]=nums[1];
12    }
13    else{
14        v[0]=nums[1];
15        v[1]=nums[0];
16    }
17    for(unsigned int i=2;i<nums.size();i++){
18        if(v[0]<nums[i]&&nums[i]<v[1]){
19            return true;
20        }
21        if(nums[i]<=v[0]){
22            de.push_front(nums[i]);
23        }
24        else if(v[1]<=nums[i]){
25            de.push_back(nums[i]);
26        }
27        if(de.size()==2){
28            v[0]=de[0];
29            v[1]=de[1];
30            de.clear();
31        }
32    }
33    return false;
34 }
```

	Test	Expected	Got	
✓	vector<int> nums {1, 10, 0, 3, 3}; cout << tripleIndexSLM(nums);	1	1	✓
✓	vector<int> nums {}; cout << tripleIndexSLM(nums);	0	0	✓
✓	vector<int> nums {15815, 15258, 10488, 9802, 5861, 3992, 2481, 1331, 550, 539, -1392, -3572, -5180, -7837, -10398, -10764, -12797, -13580, -13724, -14400}; cout << tripleIndexSLM(nums);	0	0	✓
✓	vector<int> nums {15953, 15605, 15178, 14780, 14719, 14581, 14512, 14312, 13802, 13337, 13250, 13205, 13178, 12643, 12027, 11679, 11497, 10766, 10686, 10486, 10480, 10036, 9483, 9319, 9304, 8981, 8882, 8684, 8681, 8651, 8541, 8392, 8241, 8241, 8145, 7686, 7686, 7447, 7222, 7200, 6754, 6284, 5692, 4977, 4146, 3992, 3704, 3439, 3254, 2699, 1674, 1513, 1087, 631, 317, -826, -922, -1265, -1716, -1826, -1881, -2192, -2488, -2810, -2824, -2867, -3331, -4502, -4596, -5116, -5379, -6051, -6545, -6639, -6831, -7048, -7501, -7537, -8500, -8879, -9077, -9402, -9694, -9964, -10295, -11299, -12285, -12522, -12635, -12818, -13497, -13784, -14059, -14276, -14352, -14566, -15181, -15374, -15628, -15858}; cout << tripleIndexSLM(nums);	0	0	✓
✓	vector<int> nums {16166, 16125, 16094, 16061, 16054, 15970, 15932, 15918, 15883, 15701, 15685, 15631, 15579, 15559, 15548, 15513, 15512, 15400, 15322, 15307, 15027, 14973, 14843, 14698, 14671, 14653, 14579, 14546, 14325, 14289, 14099, 14051, 13949, 13938, 13903, 13594, 13593, 13580, 13528, 13423, 13174, 13088, 13010, 13007, 12994, 12961, 12935, 12878, 12846, 12768, 12724, 12722, 12660, 12650, 12578, 12498, 12494, 12434, 12411, 12296, 12108, 11844, 11840, 11836, 11815, 11801, 11725, 11654, 11465, 11436, 11390, 11253, 11236, 11169, 11148, 11103, 11088, 11056, 11001, 10913, 10867, 10764, 10745, 10739, 10633, 10606, 10586, 10556, 10504, 10485, 10445, 10439, 10410, 10281, 10254, 10219, 10126, 10098, 9973, 9921, 9827, 9268, 9257, 9227, 9122, 9058, 9041, 9021, 8989, 8871, 8796, 8717, 8553, 8532, 8529, 8484, 8475, 8386, 8343, 8256, 8255, 8235, 8214, 8210, 8150, 8075, 7938, 7775, 7707, 7658, 7637, 7512, 7412, 7397, 7326, 7234, 7167, 7155, 7152, 7106, 7101, 7067, 6917, 6898, 6865, 6748, 6586, 6400, 6203, 6132, 6093, 5952, 5951, 5896, 5854, 5852, 5804, 5788, 5724, 5696, 5646, 5579, 5387, 5346, 5335, 5301, 5124, 5079, 5037, 4860, 4789, 4696, 4645, 4634, 4633, 4532, 4510, 4483, 4339, 4335, 4290, 4265, 4203, 3964, 3829, 3795, 3768, 3657, 3634, 3629, 3577, 3576, 3557, 3301, 3275, 3171, 3016, 2999, 2940, 2847, 2782, 2613, 2452, 2326, 2285, 2276, 2168, 2006, 1997, 1956, 1955, 1943, 1906, 1719, 1665, 1612, 1566, 1376, 1306, 1171, 1061, 1039, 978, 883, 859, 852, 819, 804, 803, 576, 566, 407, 242, 233, 229, 197, 192, 112, 37, -4, -96, -97, -110, -114, -137, -141, -216, -290, -311, -332, -378, -432, -690, -691, -746, -766, -820, -848, -853, -893, -915, -1029, -1129, -1131, -1509, -1602, -1647, -1689, -1710, -1742, -1807, -1847, -1853, -1911, -1977, -2017, -2028, -2078, -2171, -2186, -2204, -2227, -2284, -2374, -2474, -2493, -2590, -2641, -2768, -2823, -2895, -2970, -3033, -3035, -3118, -3203, -3371, -3417, -3505, -3563, -3734, -3749, -3779, -3823, -4030, -4189, -4194, -4272, -4317, -4394, -4443, -4495, -4647, -4650, -4654, -4704, -4737, -4742, -4880, -4887, -4913, -4979, -4994, -5067, -5101, -5313, -5361, -5375, -5393, -5569, -5682, -5708, -5737, -5810, -5813, -5814, -5895, -5935, -5946, -5956, -6028, -6175, -6178, -6262, -6271, -6329, -6349, -6373, -6405, -6442, -6479, -6623, -6624, -6671, -6753, -6817, -6821, -6853, -6861, -6892, -6954, -6964, -6979, -7056, -7066, -7096, -7261, -7289, -7328, -7331, -7345, -7419, -7486, -7511, -7661, -7734, -7752, -7770, -7820, -7847, -7893, -7956, -7956, -8555, -8567, -8765, -8834, -8943, -8977, -8996, -9092, -9122, -9141, -9174, -9466, -9588, -9597, -9687, -9802, -9807, -9832, -9863, -10015, -10142, -10155, -10214, -10259, -10264, -10544, -10581, -10672, -10708, -10735, -10741, -10810, -10899, -10902, -10932, -10994, -10996, -11008, -11008, -11130, -11178, -11363, -11380, -11720, -11770, -11773, -11792, -11975, -11986, -12004, -12011, -12153, -12203, -12246, -12272, -12318, -12343, -12345, -12370, -12529, -12615, -12798, -12964, -12975, -12977, -12977, -12992, -13039, -13047, -13108, -13202, -13210, -13253, -13333, -13336, -13552, -13553, -13569, -13592, -13631, -13707, -13733, -13981, -14064, -14079, -14101, -14289, -14332, -14475, -14503, -14527, -14667, -14740, -14817, -14820, -14892, -15011, -15058, -15079, -15108, -15169, -15217, -15271, -15334, -15357, -15515, -15589, -15594, -15681, -15804, -15962, -15997, -16056, -16214, -16256, -16317}; cout << tripleIndexSLM(nums);	0	0	✓
✓	vector<int> nums {16376, 16371, 16370, 16370, 16322, 16320, 16316, 16314, 16307, 16285, 16277, 16271, 16270, 16269, 16268, 16256, 16238, 16230, 16224, 16217, 16198, 16196, 16186, 16184, 16182, 16173, 16171, 16164, 16134, 16132, 16118, 16091, 16078, 16074, 16049, 16038, 16022, 16011, 16010, 16008, 16000, 15998, 15994, 15989, 15987, 15982, 15968, 15967, 15956, 15948, 15946, 15939, 15933, 15929, 15923, 15919, 15919, 15918, 15913, 15907, 15900, 15897, 15896, 15895, 15893, 15892, 15886, 15879, 15871, 15870, 15863, 15856, 15853, 15850, 15843, 15831, 15826, 15819, 15800, 15780}; cout << tripleIndexSLM(nums);	0	0	✓

Test	Expected	Got
15759, 15757, 15753, 15748, 15745, 15741, 15741, 15731, 15723, 15719, 15692, 15672, 15667, 15666, 15659, 15658, 15654, 15645, 15643, 15619, 15617, 15616, 15613, 15608, 15600, 15597, 15597, 15590, 15583, 15561, 15549, 15546, 15545, 15523, 15521, 15516, 15513, 15510, 15508, 15507, 15505, 15462, 15456, 15453, 15447, 15439, 15438, 15419, 15409, 15406, 15400, 15388, 15378, 15376, 15338, 15328, 15319, 15316, 15306, 15291, 15289, 15288, 15286, 15278, 15274, 15272, 15268, 15261, 15257, 15249, 15231, 15231, 15227, 15226, 15217, 15214, 15213, 15211, 15208, 15204, 15201, 15199, 15198, 15186, 15177, 15152, 15151, 15143, 15140, 15131, 15123, 15120, 15117, 15114, 15114, 15108, 15107, 15101, 15100, 15096, 15091, 15089, 15084, 15081, 15075, 15072, 15067, 15049, 15031, 15021, 15014, 15013, 15011, 14998, 14986, 14982, 14979, 14968, 14967, 14962, 14954, 14944, 14934, 14923, 14923, 14915, 14903, 14894, 14884, 14855, 14852, 14841, 14835, 14818, 14814, 14808, 14807, 14803, 14802, 14787, 14784, 14763, 14754, 14751, 14747, 14745, 14732, 14731, 14731, 14727, 14726, 14713, 14712, 14695, 14682, 14681, 14679, 14662, 14658, 14654, 14639, 14638, 14614, 14611, 14605, 14602, 14602, 14596, 14579, 14578, 14571, 14564, 14558, 14555, 14554, 14537, 14528, 14525, 14521, 14518, 14518, 14509, 14499, 14494, 14472, 14465, 14463, 14461, 14460, 14453, 14452, 14451, 14443, 14442, 14439, 14436, 14435, 14427, 14416, 14416, 14414, 14410, 14399, 14384, 14383, 14379, 14377, 14375, 14374, 14367, 14342, 14335, 14330, 14320, 14310, 14309, 14299, 14297, 14292, 14282, 14278, 14245, 14233, 14228, 14227, 14221, 14216, 14214, 14196, 14178, 14168, 14156, 14155, 14150, 14147, 14146, 14138, 14135, 14117, 14109, 14104, 14077, 14058, 14058, 14031, 14027, 14016, 14008, 14008, 14006, 13998, 13979, 13979, 13975, 13971, 13969, 13967, 13962, 13954, 13944, 13930, 13923, 13911, 13909, 13893, 13880, 13878, 13875, 13872, 13868, 13864, 13858, 13848, 13839, 13819, 13816, 13806, 13794, 13791, 13780, 13761, 13753, 13746, 13738, 13728, 13726, 13718, 13717, 13703, 13698, 13687, 13682, 13678, 13677, 13675, 13674, 13670, 13662, 13632, 13623, 13619, 13613, 13607, 13603, 13591, 13585, 13574, 13561, 13552, 13551, 13549, 13545, 13535, 13529, 13512, 13511, 13501, 13500, 13498, 13498, 13472, 13469, 13459, 13449, 13446, 13444, 13442, 13438, 13432, 13429, 13424, 13422, 13418, 13411, 13407, 13407, 13406, 13402, 13386, 13382, 13373, 13373, 13364, 13362, 13357, 13357, 13352, 13352, 13347, 13346, 13338, 13338, 13333, 13332, 13311, 13293, 13285, 13283, 13270, 13268, 13240, 13237, 13235, 13225, 13215, 13194, 13194, 13187, 13183, 13179, 13168, 13162, 13162, 13154, 13143, 13140, 13136, 13129, 13121, 13108, 13104, 13095, 13092, 13091, 13087, 13078, 13053, 13052, 13051, 13048, 13046, 13035, 13033, 13026, 13024, 13023, 13022, 13016, 13016, 13004, 13004, 12994, 12975, 12973, 12969, 12967, 12965, 12965, 12954, 12951, 12946, 12944, 12941, 12938, 12934, 12927, 12926, 12926, 12902, 12900, 12887, 12877, 12859, 12846, 12835, 12833, 12822, 12788, 12780, 12769, 12766, 12766, 12747, 12745, 12741, 12739, 12737, 12734, 12729, 12727, 12723, 12720, 12714, 12707, 12706, 12677, 12674, 12667, 12656, 12647, 12644, 12628, 12628, 12627, 12617, 12614, 12613, 12601, 12599, 12592, 12588, 12583, 12583, 12580, 12579, 12574, 12566, 12560, 12532, 12532, 12523, 12520, 12491, 12485, 12484, 12481, 12424, 12423, 12400, 12396, 12385, "...snip... 2, -13264, -13287, -13293, -13295, -13314, -13321, -13328, -13330, -13333, -13334, -13336, -13342, -13349, -13352, -13365, -13373, -13373, -13374, -13377, -13383, -13409, -13413, -13413, -13414, -13427, -13428, -13434, -13440, -13440, -13446, -13446, -13459, -13459, -13460, -13466, -13472, -13481, -13481, -13482, -13493, -13515, -13521, -13526, -13527, -13535, -13539, -13539, -13540, -13553, -13555, -13555, -13558, -13559, -13569, -13571, -13579, -13597, -13599, -13604, -13608, -13623, -13631, -13644, -13646, -13653, -13663, -13665, -13672, -13689, -13694, -13719, -13720, -13747, -13762, -13771, -13785, -13806, -13808, -13829, -13832, -13834, -13835, -13847, -13856, -13862, -13863, -13864, -13867, -13875, -13879, -13886, -13888, -13893, -13896, -13900, -13900, -13913, -13921, -13926, -13927, -13936, -13936, -13947, -13955, -13959, -13977, -13989, -13990, -13998, -14005, -14014, -14014, -14018, -14018, -14019, -14019, -14022, -14022, -14022, -14040, -14041, -14045, -14056, -14081, -14083, -14090, -14090, -14097, -14097, -14100, -14102, -14104, -14116, -14123, -14128, -14135, -14137, -14138, -14142, -14144, -14148, -14149, -14150, -14159, -14160, -14161, -14173, -14178, -14189, -14193, -14197, -14201, -14204, -14205, -14211, -14212, -14226, -14230, -14236, -14239, -14240, -14240, -14249, -14268, -14276, -14279, -14286, -14291, -14293, -14299, -14316, -14324, -14329, -14348, -14360, -14365, -14366, -14379, -14417, -14423, -14424, -14430, -14431, -14433, -14436, -14466, -14477, -14488, -14496, -14503, -14505, -14515, -14517, -14517, -14519, -14523, -14545, -14561, -14568, -14568, -14568, -14569, -14570, -14582, -14587, -14594, -14599, -14599, -14604, -14605, -14621, -14623, -14626, -14628, -14633, -14635, -14638, -14644, -14649, -14656, -14662, -14663, -14669, -14670, -14672, -14675, -14680, -14685, -14689, -14717, -14719, -14720, -14723, -14731, -14732, -14735, -14737, -14750, -14752, -14761, -14767, -14779, -14787, -14788, -14797, -14801, -14809, -14810, -14829, -14831, -14834, -14838, -14841, -14863, -14870, -14872, -14882, -14883, -14884, -14886, -14890, -14892, -14899, -14900, -14914, -14918, -14918, -14919, -14922, -14926, -14945, -14946, -14947, -14949, -14957, -14958, -14959, -14970, -14970, -14975, -14976, -14987, -14992, -15009, -15018, -15036, -15037, -15037, -15046, -15047, -15047, -15080, -15084, -15093, -15098, -15106, -15107, -15108, -15108, -15112, -15119, -15123, -15130, -15131, -15132, -15141, -15144, -15148, -15150, -15154, -15165, -15176, -15180, -15181, -15181, -15198, -15204, -15217, -15221, -15225, -15254, -15256, -15258, -15264, -15264, -15273, -15279, -15294, -15300, -15304, -15313, -15325, -15333, -15351, -15352, -15357, -15360, -15360, -15361, -15372, -15390, -15402, -15402, -15409, -15412, -15418, -15419, -15422, -15422, -15435, -15447, -15451, -15451, -15459, -15459, -15460, -15472, -15479,		

Test		Expected	Got
	<pre> -15480, -15484, -15489, -15496, -15499, -15506, -15515, -15541, -15544, -15544, -15549, -15551, -15556, -15576, -15588, -15589, -15591, -15592, -15599, -15602, -15603, -15605, -15606, -15614, -15624, -15625, -15626, -15630, -15632, -15636, -15641, -15647, -15652, -15653, -15663, -15675, -15678, -15695, -15706, -15722, -15730, -15733, -15735, -15760, -15761, -15772, -15774, -15775, -15782, -15785, -15787, -15788, -15789, -15790, -15817, -15831, -15841, -15843, -15844, -15872, -15882, -15886, -15893, -15894, -15903, -15911, -15926, -15929, -15933, -15962, -15963, -15970, -15972, -15974, -15993, -15995, -16002, -16016, -16022, -16023, -16026, -16030, -16044, -16046, -16052, -16061, -16099, -16114, -16124, -16128, -16129, -16131, -16136, -16155, -16160, -16161, -16163, -16169, -16175, -16176, -16179, -16180, -16182, -16202, -16202, -16202, -16208, -16215, -16218, -16224, -16227, -16244, -16249, -16259, -16265, -16285, -16292, -16293, -16298, -16300, -16308, -16319, -16327, -16334, -16346, -16347, -16350, -16352, -16364, -16366, -16381}; cout << tripleIndexSLM(nums); </pre>		
✓	<pre> vector<int> nums {15851, 2945, 13368, 4635, 2829, -10155, 14417, 12366, -3530, 6611, -9685, 13835, -15258, 2382, 10457, 1250, -3970, -8332, 10854, 14981, -12812, -15315, 7203, 16336, -53, -2618, 3392, -14074, 963, -5298, 1946, 11102, -15923, -2496, 15060, 11198, 12225, 10083, -8791, 10950, -8877, -1297, 14720, -11760, -956, 4081, 5185, 9910, 14549, -4368, -5016, 9493, -2431, 9169, 15192, -13871, -15125, 12024, 12785, 11290, 8470, 4528, 9060, 3373, -10540, -1587, 4678, 285, 5761, -12144, 5092, 5331, 4848, 12490, -553, 5459, 15270, 2116, 4568, -9150, -6330, -6049, 7125, -14939, 589, -13832, 13892, 13036, 5631, -14194, 10423, 16175, 15266, 6311, 5296, 2231, -12311, 15727, -6985, 13248, -1357, -15965, 12715, 2685, -8458, -2131, 4694, -8611, 1781, 1297, 2371, 3725, -8006, 15623, 1236, -10306, -8512, 3689, 9615, -11992, 4336, 2256, 16057, 15055, -13874, 15611, -15066, 261, 9205, -912, 8271, 6448, -1087, 15461, 4517, 7117, 12619, 8160, 9765, 16309, -14694, -4834, -6611, 6815, 11799, -3534, 14222, -13593, -8894, 7406, 9698, -4466, -2140, -13818, 1312, 2945, 9732, 3955, 8813, -13026, -4642, 1558, -2251, 15092, 7441, -10594, 4731, -13110, 7640, 3499, 11373, -29, -14597, -1229, 6204, -11903, -953, 8486, 2090, -15704, 13115, -2739, -10267, -14044, 12292, -1416, -12013, -11826, 3756, -2856, 7726, 14346, -4978, 15786, -8302, -10604, -14835, -1515, 5844, 5501, -9083, 14678, -9130, 6512, 14466, 15898, 4888, -1990, 7216, -2261, 3793, 5769, -1517, -5297, -8129, 4852, -15368, 7278, 15291, -7620, 9772, 9714, 11792, 210, 747, 6771, 5005, -8807, -15845, -7020, 1369, -7700, -14021, 12072, 12551, 15075, 12182, 6298, -12694, -14382, 700, 13707, 10155, -11757, 12082, 9916, -6248, -3578, -7831, -14801, 8767, -15019, -3347, 12169, -13380, -9603, 14303, 10171, 4431, -2678, -16219, 10209, -14158, 4641, -7200, 3946, -5719, -14652, 9374, -16006, -14886, -4782, -14126, 12684, -3361, 13517, 13313, 9052, 6810, 96, -9667, 8708, 9672, 14241, -8361, -3483, -2855, -5101, 13905, 1224, 6912, -2065, -10990, 10693, -12410, -1500, -12195, 3685, 2403, -11364, -3068, -15485, -1777, 2497, 11097, -11144, -11114, 12864, -12826, 12535, 10731, -13300, 9310, -4555, 11134, 10663, 14803, 9037, 11024, -12131, -13016, -6940, 14022, -3503, -2358, 9224, 3197, -7685, -10597, -2978, -10402, 11250, -8657, -643, -5198, 12839, 2875, 7061, -2825, -3598, 9802, -3244, -13773, -10478, -9806, -758, -9080, 16367, 13511, -13557, -3063, 5921, -4274, 3925, 11980, 9782, 309, 9330, -2496, 1456, -7551, 7584, -8511, -12722, 1927, 8752, 9578, -4211, -4769, 6188, -16341, -12435, -13674, -7620, 13339, 6008, 14371, 13514, 9743, -11519, -6500, -14747, -9877, -15702, 13226, -8349, 4408, 2345, 13243, 3275, 1911, 5451, 1014, -1422, -4427, -4808, -1932, 14435, 12768, 16122, -1363, -4220, -6567, -9671, 11312, -13961, 9290, -2232, 8600, 4031, 14650, -4250, -13893, -14812, -5921, -13763, -10972, -15272, 12339, -15185, 12740, 4826, 16074, 13522, -4380, 3867, 6754, -4295, 7518, 6880, 8380, 4470, 13, -9497, -9302, -10788, 8322, -14131, -9366, -11126, 7160, 7215, 2866, -10843, 15498, -12528, 13324, 6372, 12964, -16006, 8676, 13906, 696, -16280, 7903, -945, -1230, -3100, 14575, -4392, -14675, -1939, 1445, -13303, -4505, 10139, -10170, 14816, -7950, -11258, 7712, 324, -11987, -925, -10978, -1466, 371, 14832, -12133, 3494, -12526, -4589, -16268, -4199, 4607, -3267, 11873, -2103, 9416, -10987, -6779, 15006, 10441, -11309, -1981, -14404, -9565, 14057, -10156, 12749, 2794, 13033, 13590, -15488, 3870, -6493, 5159, 1489, 16093, 13197, -10189, -1140, -11946, -2711, 10762, 13836, 12033, -8927, -2058, 10057, 2958, -3338, -181, -10990, -9189, 985, 5183, -11882, -599, -487, 7164, 7465, 8145, 319, -4876, -10939, -14413, 9527, 2421, -11057, -208, 15536, -6019, 10652, 1113, 5828, -16177, 820, 3366, 988, 2630, -298, -3220, 8654, 10602, -2884, -8951, 13129, -1727, 10911, 13409, 1651, 10088, 3966, 1295, 11984, 5729, -7012, 4308, 5734, 14895, -2104, 12499, 10783, 8992, -11552, -10495, 9582, 10987, 1176, -15401, -16075, -2054, -653 ...snip... 551, 8939, -15086, 186, 10335, 7878, 15354, -13871, -1050, -4342, -6942, 4926, 8044, 8272, 11212, 9102, -8292, 11172, -14234, 11070, -7092, 11667, -5929, -11150, -11936, 4997, -1609, -1303, -4373, 13134, 12127, 8241, 7320, 255, 13412, -4073, 5193, -10494, 8542, -5790, 16381, 1126, 8686, -319, 9975, 5495, -5377, -15359, 13853, 2706, -6581, -1149, -6242, -3785, 14113, 6126, 3543, 4960, -922, 1569, -14984, 12561, -3109, -11412, -11550, -1999, -7523, 1023, -4396, 11928, -6414, 11682, -14777, -15767, 5954, 10919, 12694, 13474, 4708, -1763, -6430, 4593, -6884, 6727, 11189, -4662, -5919, 1613, -13851, 5046, -29, 2096, -13380, -7442, -11700, 5635, -15531, 2088, -3898, -1703, 8521, 2481, 62, -9222, -8564, -9796, -11544, 3290, -623, -7430, 4740, 4427, 9909, -15510, -8225, -10639, -7206, 10611, 2972, 5878, 16210, 1840, -13260, -14080, -13824, 4107, 5436, 10336, -13127, 13453, -3097, 12557, 10480, 4212, 5323, 7632, 11411, -250, -10520, 16192, -8912, -13914, 4070, 16113, -420, 9381, -6754, 8937, -10411, -4518, -7972, 9555, -1050, -7051, 7523, 13432, 7307, -15472, 15154, 12010, 8662, 1287, -2810, 12290, </pre>	1	1 ✓

Test	Expected	Got	
<pre> 5905, 12042, 32, 141, 10277, -5392, 6472, -61, 14011, -10802, -7406, 6265, -13664, 16201, -1696, 5783, -7826, -13656, -8664, -3893, -7831, 9774, 9080, -9601, 13223, -13146, -2377, 12809, -13964, 1343, -3795, 7970, 4340, 5530, -11834, -2662, 7434, -9694, 1042, 12362, 1145, 14492, 6289, 6217, 15433, 8446, 4654, -5021, -3731, 4229, 11529, 7366, 9634, -13566, 633, -4353, 5018, 15989, 11858, 7717, 7770, -15140, -9885, -13484, -12319, 14413, -12989, -11683, 11018, 3395, -2738, -10564, -11085, -6346, -3862, 7371, -10209, -9362, 6927, 821, 2713, -10728, 11014, 12460, -11686, -10854, -11664, -13857, -11091, -4060, -885, -15450, -12463, -10361, 4785, 9600, 8479, -12469, -5599, -12224, -14868, -14085, 16130, 7427, -13903, 6852, -8027, -6043, -2326, 1087, -3317, -8715, -15301, -2678, -5721, 6118, -15015, -13155, 8036, -6393, 9000, -805, 12266, -10507, -8415, -11032, -2531, -15730, -6665, -10565, -11289, 3242, 13690, -10764, 10040, -12026, 6587, -12177, 4171, 8577, -14991, 13593, -4822, 9099, -13925, 3330, 11892, 8666, 12391, 15028, -5794, -6562, 16211, -15420, -13576, -3976, 9362, -10074, -12110, -9369, 5041, 3781, -5602, 2307, 5498, 211, -6153, -6995, 3865, -13986, -9117, -11642, -15235, 2981, -12171, 7903, -9727, -343, -3338, 497, 6476, 11911, 9317, 3022, -651, -6140, 1084, 3224, -13717, 618, -7307, -13071, 13205, -1133, -16307, 10930, -10827, 6723, 12104, -4640, 2066, 2747, -10891, -5668, -14122, 10634, -10253, 16089, -14155, -3929, -6988, -12397, -9777, 15037, 14465, 16191, -14888, -12090, -14786, 3005, -12426, -2597, -7312, 6165, -7446, -6838, 7676, 15137, 14835, -2326, 7985, 9863, -2321, -8789, -16375, 14827, -1212, 13425, 12972, 5730, -770, -13021, -6858, 1577, 8122, -2562, 442, 3223, 5314, 10544, 12167, 12880, 1657, -9566, -1622, -7748, -11526, -10219, -332, -6217, -9732, -4240, -12867, -4982, -7687, -12456, 5288, -1766, -10097, -10750, 937, -9120, -3546, -15803, 2565, 1305, -4909, -1292, -12263, 9151, -1412, -8086, 4766, 10423, 5693, -9114, -15573, -3380, -14069, 9242, 4603, -15908, 8599, -8066, -15119, -15919, -7307, 12522, -39, -15917, 12670, 15332, -7695, -5187, 14195, -15817, 7336, 4045, 5859, -6075, -6235, -10081, 6990, 10653, -2549, -1188, -1654, 10960, -12036, 14383, 7782, -8831, 15196, 16120, -12146, 10940, 727, -13243, 10670, 6025, -7643, 4734, 10970, 6080, -10100, -15521, -4050, 798, 6823, -12885, -14415, -8039, 103, 8991, 12159, -1039, 5207, 6821, -216, -1656, 10276, -15669, -16301, -9101, 4700, 14451, -1369, 13009, -1674, -18084, -2325, -9125, -12269, -12927, 15723, 7997, -6525, 1584, -11362, -219, 13707, -13795, -761, 2713, -14747, 7976, 857, 4749, -9207, -9476, -13858, 53, -15957, -14454, 12313, 15640, -7547, -5194, 1963, -7834, 3798, -13233, 8563, 7307, 4651, 14661, 13817, 3076, -291, -1258, 11533, -2959, -11557, -2934, 2219, 4332, -9815, 3232, -2270, -4966, -4183, -2872, -12171, -1409, 2714}; cout << tripleIndexSLM(nums); </pre>			
✓ <pre> vector<int> nums {770, -8297, 11102, -1497, 13153, 8994, 7424, -15945, 5007, 14655, 9461, 50, 13644, 7100, 2210, 3323, -9279, -6867, 3103, 8282, 4198, 8763, -12741, -12431, -221, -12993, 9664, 1137, -8778, 10256, -7176, -14579, -7596, -6120, 8303, -5841, 3475, -5970, -3589, -5142, -13433, -10336, 14586, 12465, 12331, 4110, -8485, -13626, -11517, 6015, 12000, 1268, 3584, -3794, 11297, -4223, 9124, 15269, 2261, 2655, 4198, -13171, 8909, -8286, 2904, 16069, 10561, 8759, -1529, 9905, -12337, 9426, -9309, -7979, 12993, 15656, -4256, 11938, -12145, -6305, 6683, 5519, 15524, 15810, -12479, 13420, -4251, 6330, 237, 755, -10384, -5383, -9392, 12193, 4129, 11769, -12458, 4036, -7845, -3419, 12422, -10796, -4987, -5546, -5589, 518, 10962, -1024, 4017, -16097, 8390, 14766, -13231, 11654, -5386, -13781, -2575, 9886, 2063, 6914, -14523, 12882, 1049, -4439, 1339, 7410, -5891, 6906, 5014, -13507, 6213, 3042, 15576, -3393, -3548, -8611, 12646, 11357, -4682, 10146, 487, 5422, 1672, 15514, 6381, -3700, 9051, 3479, -2089, 1643, -9691, -5565, 15094, 7402, 13657, -15913, 10478, -7128, -8600, 13197, -11937, 9561, -11896, 855, -315, -487, -332, -14639, -99, 5375, -1036, -8588, 2762, -906, 11486, 3853, -15562, 14280, 861, -12434, 7868, 5756, 15793, 7780, 13382, -9758, -14921, -2839, 13618, -10004, 5483, 10394, 5627, 5695, -15189, -11669, 10773, -886, 9278, 3481, 14104, 13591, 2580, 8928, -15583, -13152, 13007, -3089, 8821, 6755, -9085, 15129, 2152, 7544, -326, -3251, 8062, -15261, -15428, -952, -4913, -11546, 9997, -12535, 688, -9703, -8906, -10619, -14705, -2372, -5944, -16227, -12769, -13877, 159, -9395, 6070, -8365, 11108, -11738, 8691, -2171, 13397, -8194, 9891, -6648, -8887, -14106, -795, -1132, -3602, 12380, -6515, 11709, 8278, 8130, -13113, -3828, 15026, -12017, 12285, -1708, -14755, 4549, -12550, -11197, -996, -11741, -14001, 4664, 7588, 13467, -16139, -3756, -15268, 12678, 16181, 16084, 3429, 8983, -7995, -7522, -14632, -12915, 3380, -14055, 590, 14253, 15751, 8002, 12454, -6288, 5669, 5373, 11010, -10503, -10256, 6568, -13037, 13769, -4939, -14602, 1175, -13770, 4806, -4613, -1505, 1817, -5147, -1405, 14102, 10696, -3228, 8360, -4865, -3004, 8075, 2212, -15874, -10317, 12858, -4893, -14694, 8976, -817, -8611, -1855, 2593, 6986, 8994, -2036, 6313, -5456, -7370, -12475, -6688, 16045, 6048, -2265, 7200, 6780, -4162, -2460, 10465, -3828, 8411, -15635, 10471, 12379, -11261, -14227, -6982, 7206, 9750, 14358, -3753, 15301, -4580, 1441, -14043, -8367, 15832, 13712, -3798, 7, -12128, -8009, -385, -15, -9108, 4346, 4097, 3267, -1748, -9384, 4170, 10755, 4703, -9431, -11296, 9836, 5857, -3421, -6229, 149, -9468, -2909, 9133, -306, -7965, 7083, -10771, -438, -1293, -3546, 6580, 15748, -805, 4084, 12028, 9691, -4474, -11998, -6618, -10113, -2698, -5639, -3093, 657, -6053, -965, -15979, 15494, -16156, 4521, 3439, -1843, -8235, 8136, -4230, -14927, -10406, -1819, 5729, -6528, 12186, -12713, -12113, 2029, -9840, 10244, 13888, 5043, -15338, -2777, 1525, -6277, -14525, 6736, 2988, -3328, -9737, 8364, 8816, 369, -14924, -5498, 9964, 12823, -13136, 4366, -4570, -7113, -2520, -12455, 5226, 6714, -2826, 7850, 14467, 6155, -9114, -6160, -2254, -16191, 5958, -4465, 15064, -2178, -10948, -10141, 13174, 1942, -7587, -2302, 2775, -10252, 7418, 13890, 10508, -4678, -579, -11268, 7449, 552, -6667, -15089, -530, 12223, 10287, </pre>	1	1	✓

Test	Expected	Got	
<pre> -5209, -10343, -1370, -250, 703, -1863, 14412, -6596, -7452, 13366, 6735, 2569, 1084, -5329, -8035, -16206, -7736, -3276, -2343, -15778, 8678, 11446, 256, 4902, 10266, -8773, 6988, -11413, -3059, 9430, -11944, -5524, -3626, 6231, 4230, 16151, 8057, -6721, 6558, -3296, 11181, 1399, 13751, -11183, -3613, -10559, 12891, -2015, -2173, 14318, -5914, 13703, -12679, -4398, -10961, -3824, 240, -2699, -150, -4304, -5539, -245, 15693, 8405, 8640, -14042, 3698, 7746, -9945, 3371, 6141, -15333, -10637, 9730, 894, 9916, -5794, 15890, -15079, -12337, 701, 6509, 3300, -1506, -12029, -12217, -6256, 16293, 1543, 3501, -11700, -4628, 10759, 9844, -14874, -2924, ...snip... 67, 12326, 1825, -6696, -660, 16247, 13104, -4633, -3635, 10475, -8622, 1093, -859, -8503, -1852, -15603, -12491, 3745, -2586, 7149, -3837, -13265, -9326, -1495, -5680, -7295, -9300, -14877, -14389, 5724, 11258, 11417, -10748, -14115, 4150, -11537, 5923, -7844, -13726, -11345, -10150, -8299, -6820, 12302, -3944, -4342, 3506, 5425, 0, -1885, -4937, 6895, -15463, -2152, 14807, -8231, -11961, 5520, -2521, 9847, -12294, -11765, 1600, -10283, 37, 14589, 7758, 3902, 10354, -4933, 13412, 4936, 15028, 1819, -4845, -8175, -15076, 5591, 13882, 9927, 945, 13875, 1055, -6546, 9722, 5088, -6977, 477, 7421, -12513, -8861, -2294, 1390, 3626, -14732, 7048, 544, 15823, 6544, 590, 6847, -15010, -6691, -4422, -3391, 12741, -8546, -1834, -4966, 7162, 14340, -14631, 3375, -12647, -14778, -15, 13198, -4401, 4852, -9153, -1828, 11382, -9080, -2525, -13170, 15745, 13374, -16197, 8413, 10395, 11813, -14426, 15219, 8971, -10378, 7182, -405, 1499, -6363, 4837, 15736, 7737, -9033, 14446, -3228, -8931, 339, 7424, -5187, 10796, -229, -8441, 13515, -8739, 701, 5707, -4668, 9946, 6980, 1141, -15037, -8535, -13259, 11983, 15351, 4491, 1495, -15029, -104, 15780, -2094, -1337, -3552, 3421, -3589, 12943, 12599, 4359, -14481, 10019, 12783, 15297, -6177, -4549, -15183, 10139, 9002, -503, 5496, -6917, -13519, 6078, -964, -9920, 10215, 15183, 12722, -3930, -3673, 2169, 15410, 10089, -14901, 12547, 10488, 11822, -13958, -8125, -6768, 11678, -4415, 15331, -12916, -4780, 4069, -7707, -16264, -7687, -11016, 2021, -11248, -6483, 6059, 8094, -2384, 9641, 14691, 9392, 5643, -3037, 8079, -2143, 7712, 9586, -13768, -5599, -7062, 6515, 14553, -8046, -11433, 2603, 4466, 11463, -5636, -9042, 15832, 1548, -7161, -1729, -3249, -15769, 7669, -4424, -10150, 13598, 2437, -6867, 15570, -10631, -14740, 9164, 6198, 6345, 13130, 11783, -15806, 3753, 2781, 15350, 6595, -14716, -7820, 4598, 3497, -168, 5799, -1129, 15454, 6841, 3910, -4683, 7510, -4072, -13425, 5092, -5912, -6843, 11684, -13038, 12892, -4045, 1084, -9369, -6075, -6660, 11672, -13456, 12049, 15140, -6471, -10627, -13858, 6434, 6864, -15289, 1423, -4836, -14602, 4054, -9697, -11277, -14819, -11358, 13878, 1919, 6470, -16135, 10490, 12572, -11, 9888, -8664, -15003, -2846, -13635, 8291, -12993, -7438, 7156, -10804, -16358, 9829, -559, 10962, -11405, 11857, 13414, -8372, -7267, -11697, 13189, -5691, -9790, 12404, -11440, -6890, 13093, 10115, 13502, -965, 3497, -16032, -12511, -14241, -8809, 8648, -2666, -10951, 12444, -12992, -11811, 16283, 8721, -9141, 6842, 8776, -7041, -4384, 2458, -15097, 1123, -3045, -11398, -15883, -14290, 6707, 10501, 5407, -11888, -1364, 508, -13744, -7178, -10546, 14299, 15656, -2990, -1963, 13242, 8046, -6630, 2666, 13169, 14759, 6547, -302, -5633, 6632, 12111, -11418, -211, 9448, 9460, -13373, 12910, 15705, 12096, -3262, 9482, -2695, -2742, -1237, 3963, -284, 4363, -14043, -6633, -13140, -1764, -13839, 12528, 7083, 14273, -9128, 6905, -10984, 4903, -6046, -918, -167, 2794, -2359, -7067, 12568, -12582, -9043, -6695, 15001, 4863, -12407, 11175, -1717, -6949, -2714, 6147, 2461, -6674, -15025, -3299, -13550, 3968, -13308, -4863, -5480, -7270, 10483, -2372, -6378, 12118, -10397, 14017, -15292, -15784, 6131, 7307, -8257, 16285, 3651, -10365, 7261, -5260, -6415, 13603, -10307, 6857, 6599, 10844, -8669, 10153, 7218, 3125, -12157, 3843, 6134, 15286, 12508, 7084, -1503, -14076, 2248, -15706, -4547, -11827, -8624, 5282, 6602, 7910, -5372, -5949, 11244, -15053, -14004, 9381, 5472, -12856, 3065, -14946, -844, 10573, -3041, 3908, -15688, -3417, 14631, 815, 11056, -15147, 5920, 1536, -565, 10263, 5974, -10847, 13145, -9642, -14554, 13519, 824, -2368, -12016, -4138, -6897, -16205, 11278, 7836, -9537, 5596, -10972, -14367, 5783, -13408, -14535, -13883, -11832, -1094, 9509, 11147, -6376, 1062, -13813, 15516, 1498, 15934, 13289, 9423, 4381, 12580, -915, -13849, -220, -15521, -11253, 11870, -13034, 12233, 13132, 11617, -9481, -15960, 13929, 14363, -15401, 7195, -10472, -1994, 1609}; cout << tripleIndexSLM(nums); </pre>			
✓ <pre> vector<int> nums {-1882, -1121, 4375, -13965, -1466, -12918, 2100, -13887, -15447, 8375, -10766, 2814, -50, -4365, -9099, 14348, -1508, 12769, -9368, -13250, -11858, 6394, -8722, 6516, -2322, -7981, -15140, -1657, 10085, 12190, -16091, 12007, -9263, 6900, -11626, 14466, -9096, 15058, 7875, 7400, -4163, -4639, 13377, -3923, -10511, 13050, 3605, 7183, 3435, 15337, -5267, 9177, -549, 7537, -12402, -1656, 8667, -16239, 15886, 5803, 16327, -1786, 1859, -6256, 15055, -6895, 14587, -14052, 10570, -1414, 1901, 7675, 9549, -6955, -15760, 16122, 2941, 6404, 10987, -13648, 7201, -15875, 15501, 6244, 12924, -5426, 16318, -11733, 1625, 2526, 14293, -1455, 11239, -11915, 395, -8198, -14905, -5762, 13743, -14583, 7524, -2420, 9244, -14527, 5507, 4636, -1303, 9898, 11035, 6170, -4641, 13111, -1946, -4698, -3536, -8283, -4552, 9495, -7579, 10150, 14224, 9024, 7833, -193, -1172, 10739, -795, -11921, -3870, -5180, 7707, -11759, -1683, 3343, -1353, 9987, 13701, 13885, 15378, 11400, -14478, 6978, 13785, -10005, 14433, 7274, 5054, 870, 13240, -7082, 3949, 4673, 3239, 10049, -15678, -11202, -15873, -6263, -5575, 2357, -3000, -7318, 11174, -8708, 526, 4108, 13606, -8311, 15995, -14912, -14251, 10587, -2517, 390, 1473, -11498, 13883, -8236, 3231, -7827, 2467, 13458, -15683, 15940, 8390, -7063, -10056, -1002, 5960, -7363, -8593, -5995, -13280, -1032, -1097, 12945, -18, -14966, 8529, 10011, 11662, -5842, -14823, -13846, -12051, 7662, 1151, -13058, 4123, -663, 11323, 11138, -1374, 5588, 7229, -5129, -11497, -8897, -1821, -12913, 5870, -5607, 4719, -6953, </pre>	1	1	✓

Test	Expected	Got
14482, 5367, 4024, -5223, 16310, -6255, 12576, -14335, -15387, -755, 16325, 3838, -2719, 14209, -772, 1515, 10282, -4429, 9723, -6289, 5469, -989, -8235, -4190, -14657, -14261, 8108, -3215, -8038, 15653, -7391, 1859, 15816, -15890, -13040, -1827, 4621, 9258, -1413, 8322, 1706, 4584, 9155, -12961, 1326, 2375, -8674, -11993, 9360, -7879, 4421, -4357, -4997, -12409, -2707, 12684, -7447, 4117, 11641, -4989, 6131, 4456, -11922, -4831, 9191, -13723, -729, 7714, -13412, 7515, 14154, 8936, 16130, -5052, -15238, 11733, 9725, -15800, 15054, 5625, -4236, 13955, -5758, -4115, 5797, -5112, -5907, 11774, -11589, -11482, -6477, 2958, -7006, 4074, 572, -14408, -2278, 10724, -12955, -3277, -15094, -5842, -14987, -10887, -1513, 1195, -10158, 11560, 374, 1421, -13835, -12376, 14078, 1050, 3611, -5106, 11906, 8623, -5571, -2752, 1373, -7163, -3133, -2214, -3506, -10829, 15989, -9135, 1927, 4084, -12494, -7209, -9472, 4523, 9956, -11650, 2321, 1956, -1613, 7199, -15040, -5603, -6535, 3453, -4776, 4997, -15754, 14630, -11589, 13443, -12986, 11162, 11768, -14022, -7921, 10927, -3936, -3124, -3329, 1076, -5802, -8164, -2058, 12245, -4570, 5337, 383, -14274, 280, -10824, 9989, 8068, 8964, -1280, 3908, -3392, -14169, -7604, 11773, 253, 9719, -9076, 2542, 2388, -15436, 15694, 7937, -2987, 4928, -4292, 12484, 11433, -2896, 12773, 8473, -5386, -7389, -6976, 7983, -1789, 15186, 877, -15696, 12317, -12183, 14143, -6340, 14009, -15594, 8696, -9193, -16038, -12577, -14893, 15390, -9547, 440, 11976, 512, -2285, 4565, -16117, 11434, -10504, -14915, -12074, 2051, -766, 8864, -6296, -4646, 15123, -11030, -9906, 4892, 6835, 1816, 4577, 13643, 9391, 6858, 14352, 15403, -13306, -2022, 4191, 9151, 6, -6355, -16336, -15757, 10675, 10247, -10687, 16283, 6248, 15524, 15479, -12772, 8641, 6468, -2342, -151, -1865, 12038, -11296, -15694, 13, 1208, -3602, -7080, 7148, -11794, 15900, 8611, -14646, -12207, -12231, 9399, -4720, 14138, -14321, -828, -10501, 5504, 4071, -6530, 6722, -626, 10320, -1843, -5894, 14264, -7907, 2501, 12343, -4401, 9133, -8232, -11019, 2802, 12911, 842, 4022, 11944, 420, -15386, 11116, 2604, 3431, -14608, 11160, -1784, -12053, 12634, 7604, 7653, -15611, -15814, 6607, -3934, -1046, -3716, 13943, -3421, -6087, -6444, -3109, -660, 11580, -8969, -7631, -8338, 8811, -15093, 1878, 15136, 1073, 15964, -14412, -9134, -15373, 13904, -10647, -2113, 6479, -15203, -3930, 2273, 14497, 6073, -6060, -15863, -11572, 1427, -15874, -15698, ...snip... 592, -2134, 3574, -12128, -1754, -5512, -8055, -16337, 15325, 6241, -2595, 7997, -13097, 14668, -811, 7536, 10285, -2118, -9866, 9225, 4653, -14700, 13211, -1593, 15278, -15322, -8727, -8696, 160, -12693, 10502, -8398, -2841, 11403, -2298, -4015, 13162, -3678, 8998, 514, 9062, -829, 4541, 8847, 10249, 14909, -913, -5797, -4548, 2588, -12636, 12673, -13120, -4109, 9784, -6216, -5508, -4135, -13195, -4230, 7957, 15538, 15166, -7323, -15936, 7088, 652, 13537, -9644, 8473, 8735, -4349, -5980, 10826, 11117, 8369, 13835, 14588, 16351, 12807, -6840, 5062, 16188, 10498, 13452, -12317, 14949, 8507, 9682, 10294, -444, -1186, 367, -14965, -13569, 1852, -6422, -9184, 2929, 11670, -8179, -15772, 8410, -8965, 10812, -6914, 5845, 15851, 10758, 15377, -11381, -3524, 15610, 9662, 6183, -7235, -8772, -13845, -6774, -6630, -12712, -5309, 6463, 12287, 15573, -13954, -10575, -4224, 5646, -1821, 4251, -583, -1347, -11509, -14655, -2639, 2777, -6937, -13346, 12933, -1246, -507, 10205, -1854, 10490, -15090, -5557, -12526, 8310, -2668, 269, -13797, -14155, 11018, -6322, -12380, 10864, -8940, -16196, 5369, 1235, -3838, 13556, -9071, 9355, 1174, 7994, 14063, 12577, -10738, -6546, -13626, -7034, -4770, 4411, -6048, -7953, -5442, -5486, -5056, 6370, 7780, -13737, 13575, -623, -12909, -10662, 12436, -6296, -14124, -12408, -15249, -4894, 14212, 16227, 5480, 7072, 15502, 6575, -3461, -14404, -10118, 9838, -6409, -6821, 11431, -14321, 14304, 8321, 980, -3613, -12143, 10026, -15387, -7472, -7515, -9834, 11492, -1975, 15640, -11036, 2375, 11645, 7850, 3042, 13169, 6863, 1924, -7644, 5243, 2439, -12770, -685, 51, -4153, -1184, -16184, 4709, -5952, 12808, 10073, -3750, -731, 9547, -4237, 6559, -12320, -7489, -12950, 5895, 773, -3208, 6156, -12557, -5884, 6257, 9084, 16337, -10718, 9774, 15934, -9757, 9770, -15456, 1166, -10942, -13729, 4169, -522, 10530, -3146, -2655, 14623, 2135, 11586, 10167, -9752, 1817, -1296, 6427, -12875, -12796, -12172, -6660, 6665, -414, -2941, -1394, -1704, 11181, 9283, -1874, 14870, 3664, 2816, 8393, 5140, -11346, -7648, -15878, -6132, 12133, -12967, -2702, 7547, 4564, -14559, 10525, 4782, 10203, 12646, 15747, -11488, 2017, -8576, 15084, 2051, 12316, 9828, -7737, -10467, 14860, -9190, 6732, 9173, -6327, 14193, 1889, -6514, -5249, 1777, -16198, 3635, 6071, 15889, -1979, 8211, -13608, 13226, 3781, 2544, -13459, 9388, -9284, -9646, -4440, -12528, 11407, -11487, 12455, 14605, 16101, 2675, 8695, -1905, 14928, 3491, -12713, 3813, -4817, 15011, 190, 11449, -12077, -15469, 2207, -3514, 6368, 4242, -7061, -59, -11786, -3116, -2728, -5592, -11375, -7180, 8289, -6133, 2350, 12428, 141, -13592, -10576, -6908, 3701, 6282, 7169, -12889, -7936, -10044, 7164, -11524, -5262, -5278, -13685, 8019, -9051, -1440, -14836, -15714, 917, -9126, -6, -14206, -8925, -1212, -10039, -2898, 12050, 4955, 13073, 6508, -11178, -8613, -15889, 15477, 7822, -269, 5801, -502, -3889, -16266, 12688, -2496, -2141, -301, 6699, -868, 8785, -13625, -15727, 3551, -16033, 7761, 14871, -15325, -4522, 13671, -3285, 5415, 8389, -8473, 8320, -13592, 5256, -9776, -4756, -5519, -6838, 4786, 5518, -781, 841, -10909, 8115, -4328, -6823, -14604, 16378, -1387, 7212, 15331, 9527, -12801, 10960, 16018, 13718, -6433, -14215, -9736, -383, -11768, -4223, -873, -9277, -14808, -8787, 1966, 6637, -13154, -5710, -8529, 6136, -4981, -11784, 2803, 11309, 3116, -5987, -14419, -12222, -921, -16195, 2589, -10042, -7517, -16143, -9722, 15295, -2851, 16149, 3681, 10897, 15352, 2383, -15789, 801, -12203, 76, 5725, -9157, -15196, 9933, 4563, 12593, 13050, -13725, 14479, -3899, 8227, 10614, 10470, -3255, 10553, 14988, -10789, 7173, 12149, -1848, 12691, 10713, -2490, -9807, -9207, -1244, -6240, -12234, 10755, -14108, -3433, 238, 6213, 7454, 14380, 2980, -252, 2176, -1433, 3602, 6557, -9403, 422, -15379, 13959, -6823, -5068, -6110, 9094, 445, 7197, 15200, 10786, -3794, 14145, -11306, 12376, -14335, -4696, 6718, -11889, -3246, -1515, 6170,		

Test	Expected	Got
16299, 16299, 16300, 16300, 16301, 16301, 16301, 16302, 16302, 16302, 16303, 16303, 16303, 16304, 16304, 16304, 16305, 16305, 16305, 16305, 16306, 16306, 16306, 16307, 16307, 16307, 16307, 16308, 16309, 16309, 16309, 16309, 16309, 16310, 16310, 16311, 16311, 16311, 16311, 16311, 16312, 16312, 16312, 16312, 16313, 16314, 16314, 16314, 16314, 16315, 16315, 16315, 16315, 16316, 16316, 16316, 16317, 16317, 16317, 16318, 16318, 16318, 16318, 16318, 16319, 16319, 16319, 16319, 16320, 16320, 16320, 16320, 16320, 16320, 16321, 16322, 16322, 16322, 16322, 16323, 16323, 16323, 16324, 16324, 16324, 16324, 16325, 16325, 16325, 16325, 16326, 16326, 16326, 16326, 16326, 16327, 16327, 16327, 16327, 16328, 16328, 16328, 16329, 16329, 16329, 16330, 16330, 16330, 16331, 16331, 16331, 16331, 16332, 16332, 16332, 16333, 16333, 16334, 16334, 16334, 16334, 16334, 16335, 16335, 16335, 16335, 16336, 16336, 16336, 16336, 16337, 16337, 16337, 16337, 16338, 16338, 16338, 16338, 16339, 16339, 16339, 16339, 16340, 16340, 16340, 16340, 16340, 16340, 16340, 16340, 16340, 16341, 16341, 16341, 16341, 16342, 16343, 16343, 16343, 16344, 16344, 16344, 16344, 16345, 16345, 16345, 16346, 16347, 16347, 16347, 16347, 16348, 16348, 16348, 16348, 16348, 16348, 16349, 16349, 16349, 16350, 16351, 16351, 16352, 16352, 16352, 16352, 16352, 16353, 16353, 16353, 16354, 16354, 16354, 16354, 16355, 16355, 16355, 16355, 16355, 16356, 16357, 16358, 16358, 16358, 16359, 16359, 16359, 16359, 16359, 16360, 16360, 16360, 16361, 16361, 16362, 16362, 16362, 16362, 16362, 16362, 16363, 16363, 16363, 16363, 16363, 16364, 16364, 16364, 16365, 16365, 16365, 16365, 16365, 16365, 16365, 16365, 16366, 16366, 16366, 16366, 16366, 16367, 16368, 16368, 16368, 16370, 16370, 16370, 16370, 16371, 16371, 16371, 16371, 16372, 16372, 16372, 16373, 16373, 16374, 16375, 16375, 16375, 16375, 16376, 16376, 16377, 16377, 16377, 16377, 16377, 16378, 16379, 16379, 16379, 16379, 16380, 16380, 16380, 16380, 16381, 16381, 16381, 16382, 16382, 16382, 16382, 16383, 16383}; cout << tripleIndexSLM(nums);		

Passed all tests! ✓

Chính xác

Điểm cho bài nộp này: 1.00/1.00.

Câu hỏi 15

Chính xác

Điểm 1,00 của 1,00

Given a string **s** containing just the characters '**'**', '**'**', '**[**', '**]**', '**{**', and '**}**'. Check if the input string is valid based on following rules:

1. Open brackets must be closed by the same type of brackets.
2. Open brackets must be closed in the correct order.

For example:

- String "**[]()**" is a valid string, also "**[(())]**".
- String "**[])**" is **not** a valid string.

Your task is to implement the function

```
bool isValidParentheses (string s){
    /*TODO*/
}
```

For example:

Test	Result
cout << isValidParentheses("[]");	1
cout << isValidParentheses("[]()");	1
cout << isValidParentheses("[")");	0

Answer: (penalty regime: 0, 0, 5, 10 %)

Reset answer

```
1 ✓ bool isValidParentheses (string s){
2     /*TODO*/
3     stack<char> stac;
4     for(unsigned int i=0;i<s.size();i++){
5         if(s[i]==' '||s[i]==[' '||s[i]=='{
6             stac.push(s[i]);
7         } else{
8             if(stac.size()==0) return 0;
9             if((s[i]==')' && stac.top()!='(')|||(s[i]==']'&&stac.top()!='[')|||(s[i]=='}'&&stac.top()!='{')){
10                 return 0;
11             }
12         } else {
13             stac.pop();
14         }
15     }
16     //return (stac.size()==0? 1 : 0);
17     if(stac.size()==0){
18         return 1;
19     }
20     else {
21         return 0;
22     }
23 }
24 }
```

B, IOA E-LEARNING



W

:

HCMUT



MyBK

BKSI

LIÊN HỆ

📍 268 Lý Thường Kiệt, P.14, Q.10, TP.HCM

☎ (028) 38 651 670 - (028) 38 647 256 (Ext: 5258, 5234)

✉ elearning@hcmut.edu.vn

Copyright 2007-2022 BKEL - Phát triển dựa trên Moodle