





You have answered correctly.

Assume yourself as Judge. Given a problem and program, Select the Status which would be generated when the program is submitted to you.

[Problem Description](http://cognizant.e-box.co.in/scodhvestComponent/reviewSubmission?pid=954&cid=34861" \l "conProblemDescription1455165110219)

**Minimum Scalar Product**

You are given two vectors v1=(x1,x2,...,xn) and v2=(y1,y2,...,yn). The scalar product of these vectors is a single number, calculated as x1y1+x2y2+...+xnyn.

Suppose you are allowed to permute the coordinates of each vector as you wish. Choose two permutations such that the scalar product of your two new vectors is the smallest possible, and output that minimum scalar product.

**Input Format**

The first line contains integer number n. The next two lines contain n integers each, giving the coordinates of v1 and v2 respectively.

**Output Format**

Output consists of a single integer, Y that corresponds to the minimum scalar product of all permutations of the two given vectors.

**Limits**

0 ≤ n ≤ 800

0 ≤ xi, yi ≤ 100000

**Sample Input 1:**

3

1 3 5

2 4 1

**Sample Output 1:**

15

**Sample Input 2:**

5

1 2 3 4 5

1 0 1 0 1

**Sample Output 2:**

6

[Given program](http://cognizant.e-box.co.in/scodhvestComponent/reviewSubmission?pid=954&cid=34861" \l "conGivenprogram1455165110250)

|  |  |
| --- | --- |
| 01 | using System; |
| 02 | using System.Collections.Generic; | |

|  |  |
| --- | --- |
| 03 | using System.Linq; |
| 04 | using System.Text; |

|  |  |
| --- | --- |
| 05 |  |
| 06 | namespace ConsoleApp | |

|  |  |
| --- | --- |
| 07 | { |
| 08 | class Program | |

|  |  |  |
| --- | --- | --- |
| 09 | { | |
| 10 |  |

|  |  |  |
| --- | --- | --- |
| 11 | static Stack<int> GetStack(int[] arr, int n) | |
| 12 | { |

|  |  |  |
| --- | --- | --- |
| 13 | Stack<int> st = new Stack<int>(); | |
| 14 |  |

|  |  |  |
| --- | --- | --- |
| 15 | for (int i = 0; i < n; i++) | |
| 16 | { |

|  |  |
| --- | --- |
| 17 |  |
| 18 | st.Push(arr[i]); | |

|  |  |  |
| --- | --- | --- |
| 19 | } | |
| 20 |  |

|  |  |
| --- | --- |
| 21 |  |
| 22 | return st; | |

|  |  |  |
| --- | --- | --- |
| 23 | } | |
| 24 |  |

|  |  |
| --- | --- |
| 25 |  |
| 26 |  |

|  |  |  |
| --- | --- | --- |
| 27 | static void Main(string[] args) | |
| 28 | { |

|  |  |
| --- | --- |
| 29 |  |
| 30 |  |

|  |  |  |
| --- | --- | --- |
| 31 | int n = int.Parse(Console.ReadLine()); | |
| 32 | int[] a1 = new int[n]; |

|  |  |  |
| --- | --- | --- |
| 33 | int[] a2 = new int[n]; | |
| 34 |  |

|  |  |  |
| --- | --- | --- |
| 35 | int i, ans = 0; | |
| 36 |  |

|  |  |  |
| --- | --- | --- |
| 37 | string[] s1 = Console.ReadLine().Split(' '); | |
| 38 |  |

|  |  |  |
| --- | --- | --- |
| 39 | string[] s2 = Console.ReadLine().Split(' '); | |
| 40 |  |

|  |  |
| --- | --- |
| 41 |  |
| 42 | for (i = 0; i < n; i++) | |

|  |  |  |
| --- | --- | --- |
| 43 | a1[i] = int.Parse(s1[i]); | |
| 44 | for (i = 0; i < n; i++) |

|  |  |  |
| --- | --- | --- |
| 45 | a2[i] = int.Parse(s2[i]); | |
| 46 |  |

|  |  |
| --- | --- |
| 47 | Array.Sort(a2); |
| 48 | Array.Sort(a1); |

|  |  |
| --- | --- |
| 49 |  |
| 50 |  |

|  |  |
| --- | --- |
| 51 |  |
| 52 |  |

|  |  |
| --- | --- |
| 53 |  |
| 54 |  |

|  |  |  |
| --- | --- | --- |
| 55 | Queue<int> col = new Queue<int>(); | |
| 56 |  |

|  |  |  |
| --- | --- | --- |
| 57 | for (i = 0; i < n; i++) | |
| 58 | { |

|  |  |
| --- | --- |
| 59 |  |
| 60 | col.Enqueue(a1[i]); | |

|  |  |  |
| --- | --- | --- |
| 61 | } | |
| 62 |  |

|  |  |
| --- | --- |
| 63 |  |
| 64 | var stack = GetStack(a2, n); | |

|  |  |
| --- | --- |
| 65 |  |
| 66 |  |

|  |  |
| --- | --- |
| 67 |  |
| 68 |  |

|  |  |
| --- | --- |
| 69 |  |
| 70 | for (i = 0; i < n; i++) | |

|  |  |
| --- | --- |
| 71 | { |
| 72 | ans = ans + col.Dequeue() \* stack.Pop(); | |

|  |  |  |
| --- | --- | --- |
| 73 | } | |
| 74 |  |

|  |  |  |
| --- | --- | --- |
| 75 | Console.WriteLine(ans); | |
| 76 |  |

|  |  |
| --- | --- |
| 77 |  |
| 78 |  |

|  |  |  |
| --- | --- | --- |
| 79 | Console.ReadKey(); | |
| 80 |  |

|  |  |  |
| --- | --- | --- |
| 81 | } | |
| 82 | } |

|  |  |
| --- | --- |
| 83 | } |

Option 1 - Wrong Answerhttp://cognizant.e-box.co.in/static/images/cross_red_small.gif  
  
Option 2 - Compile Time Errorhttp://cognizant.e-box.co.in/static/images/cross_red_small.gif  
  
Option 3 - Acceptedhttp://cognizant.e-box.co.in/static/images/tick_green_small.gif  
  
Option 4 - Runtime Errorhttp://cognizant.e-box.co.in/static/images/cross_red_small.gif

You have answered incorrectly.

Given a problem and program, fillup the blank spaces in the program in order to make the program work correctly.

[Problem Description](http://cognizant.e-box.co.in/scodhvestComponent/reviewSubmission?pid=966&cid=34862" \l "conProblemDescription1455165158683)

**Discrepancies in the Voters List**

As you might remember, the collector of Siruseri had ordered a complete revision of the Voters List. He knew that constructing the list of voters is a difficult task, prone to errors. Some voters may have been away on vacation, others may have moved during the enrollment and so on.

To be as accurate as possible, he entrusted the task to three different officials. Each of them was to independently record the list of voters and send it to the collector. In Siruseri, every one has a ID number and the list would only list the ID numbers of the voters and not their names. The officials were expected to arrange the ID numbers in ascending order in their lists.

On receiving the lists, the Collector realised that there were discrepancies - the three lists were not identical. He decided to go with the majority. That is, he decided to construct the final list including only those ID numbers that appeared in at least 2 out of the 3 lists. For example if the three lists were

23 30 42 57 90 21 23 35 57 90 92 21 23 30 57 90

then the final list compiled by the collector would be:

21 23 30 57 90

The ID numbers 35, 42 and 92 which appeared in only one list each do not figure in the final list.

Your task is to help the collector by writing a program that produces the final list from the three given lists.

**Input format**

The first line of the input contains 3 integers *N*1, *N*2 and *N*3. *N*1 is the number of voters in the first list, *N*2 is the number of voters in the second list and *N*3 is the number of voters in the third list. The next *N*1 lines (lines 2,...,*N*1+1) contain one positive integer each and describe the first list in ascending order. The following *N*2 lines (lines*N*1+2,...,*N*1+*N*2+1) describe the second list in ascending order and the final *N*3 lines (lines *N*1+*N*2+2,...,*N*1+*N*2+*N*3+1) describe the third list in ascending order.

Assumption:

Assume that N1, N2 and N3 are less than 10.

**Output format**

The first line of the output should contain a single integer *M* indicating the number voters in the final list. The next *M* lines (lines 2,...,*M*+1) should contain one positive integer each, describing the list of voters in the final list, in ascending order.

**Sample input:**

5 6 5 23 30 42 57 90 21 23 35 57 90 92 21 23 30 57 90

**Sample output:**

5 21 23 30 57 90

[Given program](http://cognizant.e-box.co.in/scodhvestComponent/reviewSubmission?pid=966&cid=34862" \l "conGivenprogram1455165158716)

|  |  |
| --- | --- |
| 01 | using System; |
| 02 | using System.Collections.Generic; | |

|  |  |
| --- | --- |
| 03 | using System.Linq; |
| 04 | using System.Text; |

|  |  |
| --- | --- |
| 05 | using System.Threading.Tasks; |
| 06 | namespace ConsoleApplication11 | |

|  |  |
| --- | --- |
| 07 | { |
| 08 | class Program | |

|  |  |
| --- | --- |
| 09 | { |
| 10 | static void Main(string[] args) | |

|  |  |
| --- | --- |
| 11 | { |
| 12 | string[] arr = Console.ReadLine().Split(' '); | |

|  |  |
| --- | --- |
| 13 | int n1 = int.Parse(arr[0]); |
| 14 | int n2 = int.Parse(arr[1]); |

|  |  |
| --- | --- |
| 15 | int n3 = int.Parse(arr[2]); |
| 16 | List<int> arr1 = new List<int>(); | |

|  |  |
| --- | --- |
| 17 | List<int> arr2 = new List<int>(); |
| 18 | List<int> arr3 = new List<int>(); |

|  |  |
| --- | --- |
| 19 | for (\_\_\_\_\_\_\_\_\_\_\_\_) |
| 20 | arr1.Add(int.Parse(Console.ReadLine())); | |

|  |  |
| --- | --- |
| 21 | for (\_\_\_\_\_\_\_\_\_\_\_\_) |
| 22 | arr2.Add(int.Parse(Console.ReadLine())); | |

|  |  |
| --- | --- |
| 23 | for (\_\_\_\_\_\_\_\_\_\_\_\_) |
| 24 | arr3.Add(int.Parse(Console.ReadLine())); | |

|  |  |  |
| --- | --- | --- |
| 25 | List<int> a = new List<int>(); | |
| 26 | for (\_\_\_\_\_\_\_\_\_\_\_\_) |

|  |  |
| --- | --- |
| 27 | { |
| 28 | for (\_\_\_\_\_\_\_\_\_\_\_\_) | |

|  |  |
| --- | --- |
| 29 | { |
| 30 | if (\_\_\_\_\_\_\_\_\_\_\_\_) | |

|  |  |
| --- | --- |
| 31 | { |
| 32 | a.Add(arr1[i]); | |

|  |  |  |
| --- | --- | --- |
| 33 | } | |
| 34 | } |

|  |  |
| --- | --- |
| 35 | } |
| 36 | for (\_\_\_\_\_\_\_\_\_\_\_\_) | |

|  |  |
| --- | --- |
| 37 | { |
| 38 | for (\_\_\_\_\_\_\_\_\_\_\_\_) | |

|  |  |
| --- | --- |
| 39 | { |
| 40 | if (\_\_\_\_\_\_\_\_\_\_\_\_) | |

|  |  |
| --- | --- |
| 41 | { |
| 42 | a.Add(arr1[i]); | |

|  |  |  |
| --- | --- | --- |
| 43 | } | |
| 44 | } |

|  |  |
| --- | --- |
| 45 | } |
| 46 | for (\_\_\_\_\_\_\_\_\_\_\_\_) | |

|  |  |
| --- | --- |
| 47 | { |
| 48 | for (\_\_\_\_\_\_\_\_\_\_\_\_) | |

|  |  |
| --- | --- |
| 49 | { |
| 50 | if (\_\_\_\_\_\_\_\_\_\_\_\_) | |

|  |  |
| --- | --- |
| 51 | { |
| 52 | a.Add(arr2[i]); | |

|  |  |  |
| --- | --- | --- |
| 53 | } | |
| 54 | } |

|  |  |
| --- | --- |
| 55 | } |
| 56 | a.Sort(); | |

|  |  |  |
| --- | --- | --- |
| 57 | a = a.Distinct().ToList(); | |
| 58 | if (\_\_\_\_\_\_\_\_\_\_\_\_) |

|  |  |  |
| --- | --- | --- |
| 59 | Console.WriteLine(a.Count - 1); | |
| 60 | else |

|  |  |  |
| --- | --- | --- |
| 61 | Console.WriteLine(a.Count); | |
| 62 | for (\_\_\_\_\_\_\_\_\_\_\_\_) |

|  |  |
| --- | --- |
| 63 | if (\_\_\_\_\_\_\_\_\_\_\_\_) |
| 64 | Console.WriteLine(a[i]); | |

|  |  |  |
| --- | --- | --- |
| 65 | } | |
| 66 | } |

|  |  |
| --- | --- |
| 67 | } |

[Actual program](http://cognizant.e-box.co.in/scodhvestComponent/reviewSubmission?pid=966&cid=34862" \l "conActualprogram1455165158729)

|  |  |
| --- | --- |
| 1 |  |

[Expected program](http://cognizant.e-box.co.in/scodhvestComponent/reviewSubmission?pid=966&cid=34862" \l "conExpectedprogram1455165158740)

|  |  |
| --- | --- |
| 01 | using System; |
| 02 | using System.Collections.Generic; | |

|  |  |
| --- | --- |
| 03 | using System.Linq; |
| 04 | using System.Text; |

|  |  |  |
| --- | --- | --- |
| 05 | using System.Threading.Tasks; | |
| 06 |  |

|  |  |  |
| --- | --- | --- |
| 07 | namespace ConsoleApplication11 | |
| 08 | { |

|  |  |
| --- | --- |
| 09 |  |
| 10 |  | |

|  |  |  |
| --- | --- | --- |
| 11 | class Program | |
| 12 | { |

|  |  |  |
| --- | --- | --- |
| 13 | static void Main(string[] args) | |
| 14 | { |

|  |  |  |
| --- | --- | --- |
| 15 | string[] arr = Console.ReadLine().Split(' '); | |
| 16 | int n1 = int.Parse(arr[0]); |

|  |  |
| --- | --- |
| 17 | int n2 = int.Parse(arr[1]); |
| 18 | int n3 = int.Parse(arr[2]); |

|  |  |
| --- | --- |
| 19 |  |
| 20 | List<int> arr1 = new List<int>(); | |

|  |  |
| --- | --- |
| 21 | List<int> arr2 = new List<int>(); |
| 22 | List<int> arr3 = new List<int>(); |

|  |  |
| --- | --- |
| 23 |  |
| 24 | for (int i = 0; i < n1; i++) | |

|  |  |  |
| --- | --- | --- |
| 25 | arr1.Add(int.Parse(Console.ReadLine())); | |
| 26 | for (int i = 0; i < n2; i++) |

|  |  |  |
| --- | --- | --- |
| 27 | arr2.Add(int.Parse(Console.ReadLine())); | |
| 28 | for (int i = 0; i < n3; i++) |

|  |  |  |
| --- | --- | --- |
| 29 | arr3.Add(int.Parse(Console.ReadLine())); | |
| 30 |  |

|  |  |  |
| --- | --- | --- |
| 31 | List<int> a = new List<int>(); | |
| 32 |  |

|  |  |  |
| --- | --- | --- |
| 33 | for (int i = 0; i < n1; i++) | |
| 34 | { |

|  |  |  |
| --- | --- | --- |
| 35 | for (int j = 0; j < n2; j++) | |
| 36 | { |

|  |  |  |
| --- | --- | --- |
| 37 | if (arr1[i] == arr2[j]) | |
| 38 | { |

|  |  |  |
| --- | --- | --- |
| 39 | a.Add(arr1[i]); | |
| 40 | } |

|  |  |  |
| --- | --- | --- |
| 41 | } | |
| 42 | } |

|  |  |
| --- | --- |
| 43 |  |
| 44 | for (int i = 0; i < n1; i++) | |

|  |  |
| --- | --- |
| 45 | { |
| 46 | for (int j = 0; j < n3; j++) | |

|  |  |
| --- | --- |
| 47 | { |
| 48 | if (arr1[i] == arr3[j]) | |

|  |  |
| --- | --- |
| 49 | { |
| 50 | a.Add(arr1[i]); | |

|  |  |  |
| --- | --- | --- |
| 51 | } | |
| 52 | } |

|  |  |  |
| --- | --- | --- |
| 53 | } | |
| 54 |  |

|  |  |  |
| --- | --- | --- |
| 55 | for (int i = 0; i < n2; i++) | |
| 56 | { |

|  |  |  |
| --- | --- | --- |
| 57 | for (int j = 0; j < n3; j++) | |
| 58 | { |

|  |  |  |
| --- | --- | --- |
| 59 | if (arr2[i] == arr3[j]) | |
| 60 | { |

|  |  |  |
| --- | --- | --- |
| 61 | a.Add(arr2[i]); | |
| 62 | } |

|  |  |  |
| --- | --- | --- |
| 63 | } | |
| 64 | } |

|  |  |
| --- | --- |
| 65 | a.Sort(); |
| 66 | a = a.Distinct().ToList(); | |

|  |  |
| --- | --- |
| 67 | if (a[0] == 0) |
| 68 | Console.WriteLine(a.Count - 1); | |

|  |  |
| --- | --- |
| 69 | else |
| 70 | Console.WriteLine(a.Count); | |

|  |  |  |
| --- | --- | --- |
| 71 | for (int i = 0; i < a.Count; i++) | |
| 72 | if (a[i] != 0) |

|  |  |  |
| --- | --- | --- |
| 73 | Console.WriteLine(a[i]); | |
| 74 | } |

|  |  |  |
| --- | --- | --- |
| 75 | } | |
| 76 | } |

[Failed Testcase](http://cognizant.e-box.co.in/scodhvestComponent/reviewSubmission?pid=966&cid=34862" \l "conFailedTestcase1455165158751)

You have answered incorrectly.

Given a problem and correct (shuffled) program, re-arrange the shuffled code segments in the correct order.

[Problem Description](http://cognizant.e-box.co.in/scodhvestComponent/reviewSubmission?pid=959&cid=34898" \l "conProblemDescription1455165193343)

**Parentheses Balance**

You are given a string consisting of parentheses () and [].  
Write a program that  a  stringsof this type and checks whether it forms a balanced set of parenthesis. Your program can assume that the maximum string length is 128.  
  
**Input**  
Input consists of a string of parentheses () and [].  
  
**Output**  
Print 'Yes' or 'No' depending on whether it is a balanced set of parenthesis or not.  
  
**Sample Input** 1  
([])  
  
**Sample Output** 1  
Yes  
  
**Sample Input** 2  
(([()])))  
**Sample Output** 2  
No

[Given program](http://cognizant.e-box.co.in/scodhvestComponent/reviewSubmission?pid=959&cid=34898" \l "conGivenprogram1455165193374)

|  |  |
| --- | --- |
| 01 | { |
| 02 | else | |

|  |  |
| --- | --- |
| 03 | { |
| 04 | { |

|  |  |
| --- | --- |
| 05 | { |
| 06 | if (st.Count != 0) | |

|  |  |
| --- | --- |
| 07 | { |
| 08 | } |

|  |  |  |
| --- | --- | --- |
| 09 | st.Push(s[i]); | |
| 10 | continue; |

|  |  |
| --- | --- |
| 11 | st.Pop(); |
| 12 | int flag = -88; | |

|  |  |
| --- | --- |
| 13 | } |
| 14 | flag = -1; | |

|  |  |
| --- | --- |
| 15 | { |
| 16 | } |

|  |  |  |
| --- | --- | --- |
| 17 | else | |
| 18 | { |

|  |  |  |
| --- | --- | --- |
| 19 | using System.Collections.Generic; | |
| 20 | { |

|  |  |
| --- | --- |
| 21 | { |
| 22 | { |

|  |  |
| --- | --- |
| 23 | break; |
| 24 | if (st.Count != 0) | |

|  |  |
| --- | --- |
| 25 | } |
| 26 | if (sqb == true) | |

|  |  |  |
| --- | --- | --- |
| 27 | else if (s[i] == ']' && t == '[') | |
| 28 | flag = -1; |

|  |  |
| --- | --- |
| 29 | } |
| 30 | { |

|  |  |
| --- | --- |
| 31 | } |
| 32 | sqb = false; | |

|  |  |
| --- | --- |
| 33 | else |
| 34 | sqb = true; | |

|  |  |  |
| --- | --- | --- |
| 35 | sqb = false; | |
| 36 | } |

|  |  |
| --- | --- |
| 37 | st.Pop(); |
| 38 | class Program | |

|  |  |
| --- | --- |
| 39 | if (s[i] == '(') |
| 40 | Console.WriteLine("No"); | |

|  |  |  |
| --- | --- | --- |
| 41 | if ((s[i] == ')' || s[i] == ']') && st.Any()) | |
| 42 | st.Push(s[i]); |

|  |  |
| --- | --- |
| 43 | } |
| 44 | if (s[i] == ')' && t == '(' && sqb == false) | |

|  |  |  |
| --- | --- | --- |
| 45 | break; | |
| 46 | } |

|  |  |
| --- | --- |
| 47 | } |
| 48 | { |

|  |  |
| --- | --- |
| 49 | else if (st.Any()) |
| 50 | using System.Text; | |

|  |  |
| --- | --- |
| 51 | } |
| 52 | { |

|  |  |  |
| --- | --- | --- |
| 53 | Console.WriteLine("No"); | |
| 54 | else |

|  |  |
| --- | --- |
| 55 | else |
| 56 | using System.Threading.Tasks; | |

|  |  |  |
| --- | --- | --- |
| 57 | continue; | |
| 58 | } |

|  |  |
| --- | --- |
| 59 | } |
| 60 | { |

|  |  |
| --- | --- |
| 61 | } |
| 62 | char t = st.Peek(); | |

|  |  |  |
| --- | --- | --- |
| 63 | flag = -1; | |
| 64 | } |

|  |  |  |
| --- | --- | --- |
| 65 | else if (s[i] == '[') | |
| 66 | if (flag == -1) |

|  |  |
| --- | --- |
| 67 | { |
| 68 | } |

|  |  |
| --- | --- |
| 69 | } |
| 70 | if (s[i] == '(' || s[i] == '[') | |

|  |  |
| --- | --- |
| 71 | using System; |
| 72 | using System.Linq; | |

|  |  |
| --- | --- |
| 73 | { |
| 74 | { |

|  |  |
| --- | --- |
| 75 | { |
| 76 | namespace ConsoleApplication12 | |

|  |  |
| --- | --- |
| 77 | static void Main(string[] args) |
| 78 | Stack<char> st = new Stack<char>(); | |

|  |  |
| --- | --- |
| 79 | } |
| 80 | string s = Console.ReadLine(); | |

|  |  |  |
| --- | --- | --- |
| 81 | for (int i = 0; i < s.Length; i++) | |
| 82 | Console.WriteLine("Yes"); |

|  |  |
| --- | --- |
| 83 | bool sqb = false; |

[Actual program](http://cognizant.e-box.co.in/scodhvestComponent/reviewSubmission?pid=959&cid=34898" \l "conActualprogram1455165193385)

[view source](http://cognizant.e-box.co.in/scodhvestComponent/reviewSubmission?pid=959&cid=34898#viewSource)[print](http://cognizant.e-box.co.in/scodhvestComponent/reviewSubmission?pid=959&cid=34898#printSource)[?](http://cognizant.e-box.co.in/scodhvestComponent/reviewSubmission?pid=959&cid=34898#about)

|  |  |
| --- | --- |
| 01 | using System; |
| 02 | using System.Collections.Generic; | |

|  |  |
| --- | --- |
| 03 | using System.Linq; |
| 04 | using System.Text; |

|  |  |
| --- | --- |
| 05 | using System.Threading.Tasks; |
| 06 | namespace ConsoleApplication12 | |

|  |  |
| --- | --- |
| 07 | { |
| 08 | class Program | |

|  |  |
| --- | --- |
| 09 | { |
| 10 | static void Main(string[] args) | |

|  |  |
| --- | --- |
| 11 | { |
| 12 | string s = Console.ReadLine(); | |

|  |  |
| --- | --- |
| 13 | int flag = -88; |
| 14 | Stack<char> st = new Stack<char>(); | |

|  |  |  |
| --- | --- | --- |
| 15 | for (int i = 0; i < s.Length; i++) | |
| 16 | { |

|  |  |  |
| --- | --- | --- |
| 17 | if (s[i] == '(' || s[i] == '[') | |
| 18 | { |

|  |  |  |
| --- | --- | --- |
| 19 | st.Push(s[i]); | |
| 20 | } |

|  |  |
| --- | --- |
| 21 | else |
| 22 | if (st.Count != 0) | |

|  |  |
| --- | --- |
| 23 | } |
| 24 | continue; | |

|  |  |
| --- | --- |
| 25 | st.Pop(); |
| 26 | else if (s[i] == ']' && t == '[') | |

|  |  |  |
| --- | --- | --- |
| 27 | flag = -1; | |
| 28 | { |

|  |  |
| --- | --- |
| 29 | } |
| 30 | else | |

|  |  |
| --- | --- |
| 31 | { |
| 32 | { |

|  |  |
| --- | --- |
| 33 | { |
| 34 | { |

|  |  |
| --- | --- |
| 35 | break; |
| 36 | if (st.Count != 0) | |

|  |  |  |
| --- | --- | --- |
| 37 | if ((s[i] == ')' || s[i] == ']') && st.Any()) | |
| 38 | } |

|  |  |  |
| --- | --- | --- |
| 39 | if (sqb == true) | |
| 40 | flag = -1; |

|  |  |
| --- | --- |
| 41 | } |
| 42 | { |

|  |  |
| --- | --- |
| 43 | } |
| 44 | sqb = false; | |

|  |  |
| --- | --- |
| 45 | else |
| 46 | sqb = true; | |

|  |  |
| --- | --- |
| 47 | sqb = false; |
| 48 | if (s[i] == ')' && t == '(' && sqb == false) | |

|  |  |
| --- | --- |
| 49 | } |
| 50 | st.Pop(); | |

|  |  |
| --- | --- |
| 51 | if (s[i] == '(') |
| 52 | Console.WriteLine("No"); | |

|  |  |  |
| --- | --- | --- |
| 53 | st.Push(s[i]); | |
| 54 | } |

|  |  |  |
| --- | --- | --- |
| 55 | break; | |
| 56 | } |

|  |  |
| --- | --- |
| 57 | } |
| 58 | { |

|  |  |  |
| --- | --- | --- |
| 59 | else if (st.Any()) | |
| 60 | } |

|  |  |
| --- | --- |
| 61 | { |
| 62 | { |

|  |  |  |
| --- | --- | --- |
| 63 | Console.WriteLine("No"); | |
| 64 | else |

|  |  |
| --- | --- |
| 65 | else |
| 66 | continue; | |

|  |  |
| --- | --- |
| 67 | } |
| 68 | } |

|  |  |
| --- | --- |
| 69 | } |
| 70 | char t = st.Peek(); | |

|  |  |  |
| --- | --- | --- |
| 71 | flag = -1; | |
| 72 | } |

|  |  |  |
| --- | --- | --- |
| 73 | else if (s[i] == '[') | |
| 74 | bool sqb = false; |

|  |  |  |
| --- | --- | --- |
| 75 | if (flag == -1) | |
| 76 | { |

|  |  |  |
| --- | --- | --- |
| 77 | Console.WriteLine("Yes"); | |
| 78 | } |

|  |  |
| --- | --- |
| 79 | } |
| 80 | { |

|  |  |
| --- | --- |
| 81 | { |
| 82 | { |

|  |  |
| --- | --- |
| 83 | } |

[Expected program](http://cognizant.e-box.co.in/scodhvestComponent/reviewSubmission?pid=959&cid=34898" \l "conExpectedprogram1455165193396)

|  |  |
| --- | --- |
| 01 | using System; |
| 02 | using System.Collections.Generic; | |

|  |  |
| --- | --- |
| 03 | using System.Linq; |
| 04 | using System.Text; |

|  |  |  |
| --- | --- | --- |
| 05 | using System.Threading.Tasks; | |
| 06 |  |

|  |  |  |
| --- | --- | --- |
| 07 | namespace ConsoleApplication12 | |
| 08 | { |

|  |  |
| --- | --- |
| 09 |  |
| 10 | class Program | |

|  |  |  |
| --- | --- | --- |
| 11 | { | |
| 12 |  |

|  |  |  |
| --- | --- | --- |
| 13 | static void Main(string[] args) | |
| 14 | { |

|  |  |
| --- | --- |
| 15 |  |
| 16 |  |

|  |  |
| --- | --- |
| 17 | string s = Console.ReadLine(); |
| 18 | Stack<char> st = new Stack<char>(); | |

|  |  |  |
| --- | --- | --- |
| 19 | bool sqb = false; | |
| 20 | int flag = -88; |

|  |  |  |
| --- | --- | --- |
| 21 | for (int i = 0; i < s.Length; i++) | |
| 22 | { |

|  |  |  |
| --- | --- | --- |
| 23 | if (s[i] == '(' || s[i] == '[') | |
| 24 | { |

|  |  |  |
| --- | --- | --- |
| 25 | if (s[i] == '(') | |
| 26 | { |

|  |  |
| --- | --- |
| 27 | st.Push(s[i]); |
| 28 | if (sqb == true) | |

|  |  |  |
| --- | --- | --- |
| 29 | sqb = false; | |
| 30 | } |

|  |  |  |
| --- | --- | --- |
| 31 | else if (s[i] == '[') | |
| 32 | { |

|  |  |  |
| --- | --- | --- |
| 33 | st.Push(s[i]); | |
| 34 | sqb = true; |

|  |  |  |
| --- | --- | --- |
| 35 | } | |
| 36 | } |

|  |  |  |
| --- | --- | --- |
| 37 | else | |
| 38 | { |

|  |  |  |
| --- | --- | --- |
| 39 | if ((s[i] == ')' || s[i] == ']') && st.Any()) | |
| 40 | { |

|  |  |  |
| --- | --- | --- |
| 41 | char t = st.Peek(); | |
| 42 |  |

|  |  |  |
| --- | --- | --- |
| 43 | if (s[i] == ')' && t == '(' && sqb == false) | |
| 44 | { |

|  |  |  |
| --- | --- | --- |
| 45 | if (st.Count != 0) | |
| 46 | { |

|  |  |
| --- | --- |
| 47 | st.Pop(); |
| 48 | continue; |

|  |  |
| --- | --- |
| 49 | } |
| 50 | else | |

|  |  |
| --- | --- |
| 51 | { |
| 52 | flag = -1; | |

|  |  |  |
| --- | --- | --- |
| 53 | break; | |
| 54 | } |

|  |  |
| --- | --- |
| 55 | } |
| 56 | else if (s[i] == ']' && t == '[') | |

|  |  |
| --- | --- |
| 57 | { |
| 58 | if (st.Count != 0) | |

|  |  |
| --- | --- |
| 59 | { |
| 60 | sqb = false; | |

|  |  |
| --- | --- |
| 61 | st.Pop(); |
| 62 | continue; |

|  |  |
| --- | --- |
| 63 | } |
| 64 | else | |

|  |  |
| --- | --- |
| 65 | { |
| 66 | flag = -1; | |

|  |  |  |
| --- | --- | --- |
| 67 | break; | |
| 68 | } |

|  |  |
| --- | --- |
| 69 |  |
| 70 | } | |

|  |  |
| --- | --- |
| 71 | } |
| 72 | else | |

|  |  |
| --- | --- |
| 73 | { |
| 74 | flag = -1; | |

|  |  |  |
| --- | --- | --- |
| 75 | } | |
| 76 |  |

|  |  |
| --- | --- |
| 77 |  |
| 78 | } | |

|  |  |  |
| --- | --- | --- |
| 79 | } | |
| 80 |  |

|  |  |  |
| --- | --- | --- |
| 81 | if (flag == -1) | |
| 82 | { |

|  |  |  |
| --- | --- | --- |
| 83 | Console.WriteLine("No"); | |
| 84 | } |

|  |  |  |
| --- | --- | --- |
| 85 | else if (st.Any()) | |
| 86 | { |

|  |  |  |
| --- | --- | --- |
| 87 | Console.WriteLine("No"); | |
| 88 | } |

|  |  |
| --- | --- |
| 89 | else |
| 90 | Console.WriteLine("Yes"); | |

|  |  |  |
| --- | --- | --- |
| 91 | } | |
| 92 | } |

|  |  |
| --- | --- |
| 93 | } |

[Failed Testcase](http://cognizant.e-box.co.in/scodhvestComponent/reviewSubmission?pid=959&cid=34898" \l "conFailedTestcase1455165193407)

|  |  |
| --- | --- |
| **S.NO** | **Input** |
| 1 | ([()[]()])() |
| 2 | ((([][[()]]))) |
| 3 | )( |
| 4 | (([()]))) |
| 5 | (] |
| 6 | ([]) |
| 7 | ()[](())[()]) |
| 8 | ][ |
| 9 | ( |

You have answered incorrectly.  
  
  
Your program is incorrect since it failed in one or more testcases.

Given a Program with some mistakes, correct the program with minimal compilations

[Problem Description](http://cognizant.e-box.co.in/scodhvestComponent/reviewSubmission?pid=964&cid=34912" \l "conProblemDescription1455165222599)

**Parentheses Balance**

You are given a string consisting of parentheses () and [].  
Write a program that  a  stringsof this type and checks whether it forms a balanced set of parenthesis. Your program can assume that the maximum string length is 128.  
  
**Input**  
Input consists of a string of parentheses () and [].  
  
**Output**  
Print 'Yes' or 'No' depending on whether it is a balanced set of parenthesis or not.  
  
**Sample Input** 1  
([])  
  
**Sample Output** 1  
Yes  
  
**Sample Input** 2  
(([()])))  
**Sample Output** 2  
No

[Given program](http://cognizant.e-box.co.in/scodhvestComponent/reviewSubmission?pid=964&cid=34912" \l "conGivenprogram1455165222601)

|  |  |
| --- | --- |
| 01 | using System; |
| 02 | using System.Collections.Generic; | |

|  |  |
| --- | --- |
| 03 | using System.Linq; |
| 04 | using System.Text; |

|  |  |  |
| --- | --- | --- |
| 05 | using System.Threading.Tasks; | |
| 06 |  |

|  |  |  |
| --- | --- | --- |
| 07 | namespace rails | |
| 08 | { |

|  |  |  |
| --- | --- | --- |
| 09 | class Program | |
| 10 | { |

|  |  |  |
| --- | --- | --- |
| 11 | static void Main(string[] args) | |
| 12 | { |

|  |  |
| --- | --- |
| 13 | int flag=0; |
| 14 | string a = Console.ReadLine(); | |

|  |  |  |
| --- | --- | --- |
| 15 | Stack<char> st = new Stack<char>(); | |
| 16 | for (int i = 0; i < a.Length; i++) |

|  |  |
| --- | --- |
| 17 | { |
| 18 | if (a[i] == '(' || a[i] == '[') | |

|  |  |
| --- | --- |
| 19 | { |
| 20 | st.Push(a[i]); | |

|  |  |
| --- | --- |
| 21 | } |
| 22 | else if (a[i] == ')') | |

|  |  |
| --- | --- |
| 23 | { |
| 24 | if (st.Peek() == '(') | |

|  |  |
| --- | --- |
| 25 | { |
| 26 | st.Pop(); | |

|  |  |
| --- | --- |
| 27 | } |
| 28 | else | |

|  |  |
| --- | --- |
| 29 | { |
| 30 | flag = 1; | |

|  |  |  |
| --- | --- | --- |
| 31 | break; | |
| 32 | } |

|  |  |
| --- | --- |
| 33 | } |
| 34 | else if (a[i] == ']') | |

|  |  |
| --- | --- |
| 35 | { |
| 36 | if (st.Peek()== '[') | |

|  |  |
| --- | --- |
| 37 | { |
| 38 | st.Pop(); | |

|  |  |
| --- | --- |
| 39 | } |
| 40 | else | |

|  |  |
| --- | --- |
| 41 | { |
| 42 | flag = 1; | |

|  |  |  |
| --- | --- | --- |
| 43 | break; | |
| 44 | } |

|  |  |  |
| --- | --- | --- |
| 45 | } | |
| 46 | } |

|  |  |
| --- | --- |
| 47 | if(flag==0) |
| 48 | Console.WriteLine("Yes"); | |

|  |  |
| --- | --- |
| 49 | else |
| 50 | Console.WriteLine("No"); | |

|  |  |  |
| --- | --- | --- |
| 51 | } | |
| 52 | } |

|  |  |
| --- | --- |
| 53 | } |

[Actual Program](http://cognizant.e-box.co.in/scodhvestComponent/reviewSubmission?pid=964&cid=34912" \l "conActualProgram1455165222602)

|  |  |
| --- | --- |
| 01 | using System; |
| 02 | using System.Collections.Generic; | |

|  |  |
| --- | --- |
| 03 | using System.Linq; |
| 04 | using System.Text; |

|  |  |  |
| --- | --- | --- |
| 05 | using System.Threading.Tasks; | |
| 06 |  |

|  |  |  |
| --- | --- | --- |
| 07 | namespace rails | |
| 08 | { |

|  |  |  |
| --- | --- | --- |
| 09 | class Program | |
| 10 | { |

|  |  |  |
| --- | --- | --- |
| 11 | static void Main(string[] args) | |
| 12 | { |

|  |  |
| --- | --- |
| 13 | int flag=0; |
| 14 | string a = Console.ReadLine(); | |

|  |  |  |
| --- | --- | --- |
| 15 | Stack<char> st = new Stack<char>(); | |
| 16 | for (int i = 0; i < a.Length; i++) |

|  |  |
| --- | --- |
| 17 | { |
| 18 | if (a[i] == '(' || a[i] == '[') | |

|  |  |
| --- | --- |
| 19 | { |
| 20 | st.Push(a[i]); | |

|  |  |
| --- | --- |
| 21 | } |
| 22 | else if (a[i] == ')') | |

|  |  |
| --- | --- |
| 23 | { |
| 24 | if (st.Peek() == '(') | |

|  |  |
| --- | --- |
| 25 | { |
| 26 | st.Pop(); | |

|  |  |
| --- | --- |
| 27 | } |
| 28 | else | |

|  |  |
| --- | --- |
| 29 | { |
| 30 | flag = 1; | |

|  |  |  |
| --- | --- | --- |
| 31 | break; | |
| 32 | } |

|  |  |
| --- | --- |
| 33 | } |
| 34 | else if (a[i] == ']') | |

|  |  |
| --- | --- |
| 35 | { |
| 36 | if (st.Peek()== '[') | |

|  |  |
| --- | --- |
| 37 | { |
| 38 | st.Pop(); | |

|  |  |
| --- | --- |
| 39 | } |
| 40 | else | |

|  |  |
| --- | --- |
| 41 | { |
| 42 | flag = 1; | |

|  |  |  |
| --- | --- | --- |
| 43 | break; | |
| 44 | } |

|  |  |  |
| --- | --- | --- |
| 45 | } | |
| 46 | } |

|  |  |
| --- | --- |
| 47 | if(flag==0) |
| 48 | Console.WriteLine("Yes"); | |

|  |  |
| --- | --- |
| 49 | else |
| 50 | Console.WriteLine("No"); | |

|  |  |  |
| --- | --- | --- |
| 51 | } | |
| 52 | } |

|  |  |
| --- | --- |
| 53 | } |

[Failed Inputs](http://cognizant.e-box.co.in/scodhvestComponent/reviewSubmission?pid=964&cid=34912" \l "conFailedInputs1455165222603)

|  |  |
| --- | --- |
| **S.NO** | **Input** |
| 1 | ][ |
| 2 | )( |
| 3 | (] |
| 4 | ([]) |
| 7 | ()[](())[()]) |
| 8 | ((([][[()]]))) |
| 9 | (([()]))) |