**[Minimum Remove to Make Valid Parentheses](https://leetcode.com/problems/minimum-remove-to-make-valid-parentheses/)**

Given a string s of '(' , ')' and lowercase English characters.

Your task is to remove the minimum number of parentheses ( '(' or ')', in any positions ) so that the resulting *parentheses string* is valid and return **any** valid string.

Formally, a *parentheses string* is valid if and only if:

* It is the empty string, contains only lowercase characters, or
* It can be written as AB (A concatenated with B), where A and B are valid strings, or
* It can be written as (A), where A is a valid string.

**Example 1:**

**Input:** s = "lee(t(c)o)de)"

**Output:** "lee(t(c)o)de"

**Explanation:** "lee(t(co)de)" , "lee(t(c)ode)" would also be accepted.

**Example 2:**

**Input:** s = "a)b(c)d"

**Output:** "ab(c)d"

**Example 3:**

**Input:** s = "))(("

**Output:** ""

**Explanation:** An empty string is also valid.

**Constraints:**

* 1 <= s.length <= 105
* s[i] is either'(' , ')', or lowercase English letter.

class Solution {

public:

    string minRemoveToMakeValid(string s) {

        // Initialize variables

        int openParenthesesCount = 0;

        vector<char> arr(s.begin(), s.end()); // Convert string to character array

        // First pass: mark excess closing parentheses with '\*'

        for (int i = 0; i < arr.size(); i++) {

            if (arr[i] == '(')

                openParenthesesCount++;

            else if (arr[i] == ')') {

                if (openParenthesesCount == 0)

                    arr[i] = '\*'; // Mark excess closing parentheses

                else

                    openParenthesesCount--;

            }

        }

        // Second pass: mark excess opening parentheses from the end

        for (int i = arr.size() - 1; i >= 0; i--) {

            if (openParenthesesCount > 0 && arr[i] == '(') {

                arr[i] = '\*'; // Mark excess opening parentheses

                openParenthesesCount--;

            }

        }

        // Filter out marked characters and store the result in a new string

        string result = "";

        for (char c : arr) {

            if (c != '\*')

                result += c;

        }

        return result;

    }

};

Link : <https://leetcode.com/problems/minimum-remove-to-make-valid-parentheses/?envType=daily-question&envId=2024-04-06>