

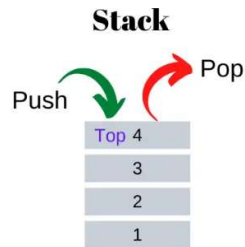
## CHUYÊN ĐỀ STACK – NGĂN XẾP

### 1. Lý thuyết về ngăn xếp(stack)

Trong khoa học máy tính, một ngăn xếp (còn gọi là bộ xếp chồng, tiếng Anh: Stack) là một cấu trúc dữ liệu trừu tượng hoạt động theo nguyên lý “vào sau ra trước”

(Last In First Out (LIFO)). Tức là, phần tử cuối cùng được chèn vào ngăn xếp sẽ là phần tử đầu tiên được lấy ra khỏi ngăn xếp.

- Thêm một cuốn sách vào hộp(push của stack)
- Lấy một cuốn sách khỏi hộp, bạn chỉ lấy được thẳng trên cùng(pop của stack)



Như vậy, việc thao tác với ngăn xếp của chúng ta chỉ bao gồm các hành động sau:

- **Push:** Thêm một phần tử vào đỉnh của ngăn xếp, số phần tử của ngăn xếp tăng lên 1.
- **Pop:** Xóa bỏ phần tử đầu tiên ở đỉnh của ngăn xếp, số phần tử của ngăn xếp giảm đi 1.
- **Top:** Lấy giá trị của phần tử đầu tiên ở đỉnh của ngăn xếp, số phần tử của ngăn xếp không thay đổi.
- **IsEmpty:** Kiểm tra ngăn xếp trống hay không. Ngăn xếp trống là ngăn xếp không có phần tử nào.
- **IsFull:** Kiểm tra ngăn xếp đã đầy hay chưa. Thao tác này không phải lúc nào cũng có.
- **Size:** Lấy số lượng phần tử stack đang có.

### 2. Cài đặt ngăn xếp bằng mảng

```
#include<bits/stdc++.h>
using namespace std;
#define max 100 // giá trị tối đa stack
struct Stack{
    int top;
    int Data[max]; //Kieu stack la so nguyen
};
Stack S; // Khai bao ngan xep S
```

#### //Khoi tao Stack

```
void Init (Stack &S){
    S.top = 0;
}
```

#### // Kiểm tra Stack có rỗng ?

```
bool IsEmpty(Stack S) {
    return S.top == 0;
}
```

#### //Kiểm tra ngăn xem đầy

```
bool IsFull(Stack S){
    return S.top == max;
```

```
}
```

### //Ham Push them phan tu x vao dau ngan xep

```
void Push(Stack & S, int x){  
    if(IsFull(S))  
        cout<<"\nNgan xep day!"<<endl;  
    else{  
        S.top ++;  
        S.Data[S.top]=x;  
    }  
}
```

### // Xoa 1 phan tu o dinh

```
void Pop(Stack & S){  
    if(IsEmpty(S))  
        cout<<"\n Ngan xep rong! "<<endl;  
    else  
        S.top--;  
}
```

### // Lay 1 pha tu o dinh

```
int Top(){  
    return S.Data[S.top];  
}
```

### // Lay so luong phu tu hien tai

```
int Size(){  
    return S.top;  
}
```

### // Chương trình chính

```
int main(){  
    int i;  
    Init(S);  
    for ( i = 10; i <= 15; i++)  
        Push(S,i);  
    cout << "Push cac phan tu 10 11 12 13 14 15 \n";  
    cout << "Tiep tuc dua 100 \n";  
    Push(S,100);  
    cout << Top() << endl;  
    Pop(S);  
    cout << "so luong pt = " << Size() <<endl;  
    cout << IsEmpty(S) << endl;  
    cout << "Lay het Stack" << endl;  
    while (!IsEmpty (S)){  
        cout <<Top() << " ";  
        Pop(S);  
    }  
}
```

## 3. Ứng dụng của stack

Chúng ta sẽ sử dụng stack vào bài toán kiểm tra dãy ngoặc có hợp lệ hay không.

Bạn có một dãy các dấu ngoặc bao gồm ngoặc đóng ‘)’ và ngoặc mở ‘(’. Bạn phải kiểm tra xem dãy ngoặc đó có hợp lệ hay không.

Một dãy ngoặc hợp lệ thì sẽ không thừa dấu ngoặc hoặc không có dấu ngoặc lẻ loi, chẳng hạn như: (), (()), ((( ))) là các dãy ngoặc hợp lệ. Còn ((, ()), (()( là các dãy ngoặc không hợp lệ.

Bạn có thể giải quyết bài toán này với stack. Hãy xem chúng ta làm như thế nào nhé.

Ý tưởng: Duyệt qua từng dấu ngoặc trong dãy ngoặc; Sử dụng một stack để *push* các dấu ngoặc mở vào stack, mỗi khi gặp dấu ngoặc đóng, thực hiện *pop* một phần tử khỏi stack. Dãy ngoặc sẽ không hợp lệ khi bạn không thể *pop* hoặc khi kết thúc duyệt mà stack vẫn chưa rỗng.

## 4. Sử dụng stack trong thư viện STL

Stack là một loại container adaptor, được thiết kế để hoạt động theo kiểu LIFO (Last -in first - out) (vào sau ra trước), tức là một kiểu danh sách mà việc bổ sung và loại bỏ một phần tử được thực hiện ở cuối danh sách. Vị trí cuối cùng của stack gọi là đỉnh (top) của ngăn xếp.

Khai báo: `#include <stack>`

- **size()** : trả về kích thước hiện tại của stack. ĐPT  $O(1)$ .
- **empty()** : true stack nếu rỗng, và ngược lại. ĐPT  $O(1)$ .
- **push(x)** : đẩy phần tử vào stack. ĐPT  $O(1)$ .
- **pop()** : loại bỏ phần tử ở đỉnh của stack. ĐPT  $O(1)$ .
- **top()** : truy cập tới phần tử ở đỉnh stack. ĐPT  $O(1)$ .
- **swap()**: Hoán đổi 2 stack. ĐPT  $O(n)$ . Hoặc có thể dùng thêm biến tg có cùng kiểu stack để hoán đổi như biến thông thường.

Cú pháp: **stack <kiểu dữ liệu> tên stack;**

Ví dụ: Khởi tạo một stack có kiểu dữ liệu là số nguyên int.      **stack <int> st;**

Chương trình demo:

```
#include <iostream>
#include <stack>
using namespace std;
stack <int> s;
int i;
main() {
    for (i=1;i<=5;i++) s.push(i); // s={1,2,3,4,5}
    s.push(100); // s={1,2,3,4,5,100}
    cout << s.top() << endl; // In ra 100
    s.pop(); // s={1,2,3,4,5}
    cout << s.empty() << endl; // In ra 0
    cout << s.size() << endl; // In ra 5
    system("pause");
}
```

## 5. Bài tập

### Bài 1. INNGUOC.CPP

Nhập vào một số N và một dãy gồm N số nguyên. Yêu cầu hãy in ra dãy đó theo thứ tự ngược lại.

Dữ liệu vào: đọc từ file INNGUOC.INP gồm:

+ Dòng đầu tiên gồm số nguyên dương N ( $N \leq 10^6$ )

+ Dòng tiếp theo gồm N số nguyên dương trong phạm vi int.

Kết quả: Ghi ra tập tin INNGUOC.OUT gồm N dòng, mỗi dòng gồm 1 số là kết quả bài toán.

INNGUOC.INP	INNGUOC.OUT
7	3

1 3 5 4 6 10 3	10
	6
	4
	5
	3
	1

## Bài 2. ROBOT.CPP

Robot VG bản update 2.0 là robot tân tiến nhất thế giới hiện tại. Các robot được lập trình một ngôn ngữ riêng để giao tiếp với nhau. Để mã hóa, chúng quyết định sẽ nói ngược lại những gì con người giao tiếp. Ví dụ con người nói: “I am human” thì khi chuyển sang ngôn ngữ của Robot đó là “namuhmaI”. Lí do để tiết kiệm tài nguyên khi giao tiếp, các con robot sẽ tự động bỏ dấu cách. Hãy chuyển các câu sau theo ngôn ngữ của robot nhé.

Dữ liệu vào: đọc từ file ROBOT.INP một dòng duy nhất là câu nói của loài người (độ dài tối đa  $10^3$  ký tự)

Kết quả: ghi ra tập tin ROBOT.OUT một dòng duy nhất là ngôn ngữ của robot

ROBOT.INP	ROBOT.OUT
I am human	namuhmaI

## Bài 3. DAUNGOAC.CPP

Hôm nay đi học, Bờm được dạy về những dấu ngoặc mở: “(” và đóng “)”. Bài tập về nhà của cô giáo là kiểm tra một dãy dấu ngoặc có hợp lệ hay không?

Một dãy dấu ngoặc hợp lệ khi:

+ Xâu rỗng là 1 dãy dấu ngoặc hợp lệ.

+ (): là một dãy dấu ngoặc hợp lệ.

+ Nếu A và B là dãy dấu ngoặc hợp lệ thì (A) và AB cũng là 2 dãy dấu ngoặc hợp lệ.

Vì mãi nói chuyện, bờm đã quên mất gợi ý của cô. Em hãy giúp Bờm nhé.

Dữ liệu vào: đọc từ file DAUNGOAC.INP gồm:

- Dòng đầu số N là độ dài của dãy dấu ngoặc của biểu thức (N chẵn,  $1 \leq n \leq 10^3$ )
- Dòng tiếp theo là dãy dấu ngoặc.

Kết quả: ghi ra file DAUNGOAC.OUT dãy ngoặc đúng YES, dãy ngoặc sai ghi NO.

DAUNGOAC.INP	DAUNGOAC.OUT	DAUNGOAC.INP	DAUNGOAC.OUT
4	YES	6	NO
(())		()()()	

## Bài 1: DauNgoac1

Cho các đoạn văn chứa các dấu ngoặc, có thể là ngoặc đơn đơn ( “()” ). Một đoạn văn đúng là đoạn mà với mỗi dấu mở ngoặc thì sẽ có dấu đóng ngoặc tương ứng và đúng thứ tự. Nhiệm vụ của bạn kiểm tra xem đoạn văn có đúng hay không.

Input

Gồm nhiều bộ test, mỗi bộ test trên một dòng chứa đoạn văn cần kiểm tra có thể bao gồm: các kí tự trong bảng chữ cái tiếng Anh, dấu cách, và dấu ngoặc (ngoặc đơn hoặc ngoặc vuông). Kết thúc mỗi bộ test là một dấu chấm. Mỗi dòng có không quá 100 kí tự.

Dữ liệu kết thúc bởi dòng chứa duy nhất một dấu chấm.

Output

Với mỗi bộ test, xuất ra trên một dòng “yes” nếu đoạn văn đúng, ngược lại in ra “no”.

Input	Output
( ( ( ( [ ] ) ) ) )	Yes
( [ ( ( ( [ ] ) ( ) ( ) ) ) ) ]	No

## Bài 2: DauNgoac2 - Đề bài PTIT123J PTIT spoj

Cho các đoạn văn chứa các dấu ngoặc, có thể là ngoặc đơn đơn ( “()” ) hoặc ngoặc vuông ( “[ ]” ). Một đoạn văn đúng là đoạn mà với mỗi dấu mở ngoặc thì sẽ có dấu đóng ngoặc tương ứng và đúng thứ tự. Nhiệm vụ của bạn kiểm tra xem đoạn văn có đúng hay không.

Input

Gồm nhiều bộ test, mỗi bộ test trên một dòng chứa đoạn văn cần kiểm tra có thể bao gồm: các kí tự trong bảng chữ cái tiếng Anh, dấu cách, và dấu ngoặc (ngoặc đơn hoặc ngoặc vuông). Kết thúc mỗi bộ test là một dấu chấm. Mỗi dòng có không quá 100 kí tự.

Dữ liệu kết thúc bởi dòng chứa duy nhất một dấu chấm.

Output

Với mỗi bộ test, xuất ra trên một dòng “yes” nếu đoạn văn đúng, ngược lại in ra “no”.

Input	Output
So when I die (the [first] I will see in (heaven) is a score list).	yes
[ first in ] ( first out ).	yes
Half Moon tonight (At least it is better than no Moon at all).	no
A rope may form )( a trail in a maze.	no
Help( I[m being held prisoner in a fortune cookie factory)].	yes
([ (([ ( [ ] ) ( ) ( ( ) ] ) ) ]).	yes
.	
.	

### Bài 3: STBRACKET – Liệt kê cặp dấu ngoặc

Cho dãy ngoặc đúng gồm dấu mở ngoặc ( và dấu đóng ngoặc ). Các dấu ngoặc được đánh số thứ tự từ đến . Hãy liệt kê chỉ số của các cặp dấu mở ngoặc và đóng ngoặc tương ứng.

#### Dữ liệu vào:

Gồm một dòng duy nhất chứa xâu ký tự biểu diễn dãy ngoặc.

#### Dữ liệu ra:

Gồm dòng, mỗi dòng gồm hai số là chỉ số của các cặp ngoặc tương ứng nhau, thứ tự liệt kê tăng dần theo chỉ số của dấu ngoặc đóng.

**Giới hạn:**  $11 \leq n \leq 10^5$

STBRACKET.inp	STBRACKET.out
0(00)	1 2 4 5 6 7 3 8

### Bài 4: STQUERY – Truy vấn với ngăn xếp

Bạn được cho một ngăn xếp rỗng và một số truy vấn với ngăn xếp này. Các truy vấn là những truy vấn cơ bản của ngăn xếp: Đẩy vào, lấy ra, in ra phần tử ở đỉnh, các truy vấn có dạng:

- 1: n: Đẩy số nguyên n vào ngăn xếp
  - 2:2: Loại bỏ phần tử ở đầu ngăn xếp (nếu ngăn xếp rỗng thì thao tác này không có hiệu lực)
  - 3:3: In ra phần tử ở đỉnh ngăn xếp (không lấy ra khỏi ngăn xếp, nếu ngăn xếp rỗng thì in ra **Empty!**)
- Dữ liệu vào
- Dòng đầu chứa số nguyên dương T là số truy vấn;
  - TT dòng tiếp theo, mỗi dòng chứa một truy vấn.

#### Giới hạn:

- $1 \leq T \leq 10^5$ ;  $|n| \leq 10^9$

Dữ liệu ra

- Ứng với mỗi truy vấn loại 3, in ra kết quả tương ứng trên một dòng.

STQUERY.inp	STQUERY.out
6 1 15 1 20 2 3 2 3	15 Empty!

## Bài 5. CHO Nguyên Tố Hóa học

Hóa chất chỉ gồm các nguyên tố C, H, O có trọng lượng 12,1, 16 tương ứng.

Nó được biểu diễn dạng “nén”, ví dụ COOHHH là CO<sub>2</sub>H<sub>3</sub> hay CH (CO<sub>2</sub>H) (CO<sub>2</sub>H) (CO<sub>2</sub>H) là CH(CO<sub>2</sub>H)<sub>3</sub>. Nếu ở dạng nén thì số lần lặp  $\geq 2$  và  $\leq 9$ .

Tính khối lượng hóa chất.

Input

Gồm một dòng mô tả hóa chất không quá 100 kí tự chỉ gồm C, H, O, (, ), 2,...,9.

Output

Khối lượng của hóa chất (luôn  $\leq 10000$ ).

Ví dụ:

CHO.INP	CHO.OUT
COOH	45
CH(CO <sub>2</sub> H) <sub>3</sub>	148
((CH) <sub>2</sub> (OH <sub>2</sub> H)(C(H))O) <sub>3</sub>	222

Ý tưởng: Sử dụng ngăn xếp (stack) quản lý và tính toán

- khi gặp ‘C’, ‘O’, ‘H’ cứ thế đưa vào stack các giá trị tương ứng
- Khi gặp ‘(’ đưa số 0 vào stack để đánh dấu mở ngoặc
- Khi gặp ‘)’ lấy lần lượt ra khỏi stack đến khi gặp số 0 (lấy cả số 0) ra tính tổng lại sau đó đưa ngược tổng vào stack
- Khi gặp số từ 2->9 thì lấy đỉnh stack nhân với số tương ứng
- Cuối cùng cộng tổng tất cả các giá trị trong stack ra kết quả cần tính

## Bài 6: STPAR – Street Parade

Hàng năm, cứ vào dịp mừng 2 tháng 9, Sơn La lại tổ chức đoàn xe diễu hành. Đoàn xe gồm xe đánh số từ đến diễu hành từ đường Trường Trinh đi về đường Chu Văn Thịnh. Khi đoàn xe đi vào đường Chu Văn Thịnh, tất cả các xe phải đi theo thứ tự (từ 1 đến n). Khi đoàn xe tham gia diễu hành sẽ xếp hàng trên đoạn đường Trường Chinh và chưa có thứ tự theo yêu cầu. Để sắp xếp đúng thứ tự cho đoàn xe tiến vào đường Chu Văn Thịnh, ban tổ chức sử dụng đoạn đường Điện Biên để cho các xe tránh vào đó.

Trên các đoạn đường, các xe không được phép vượt nhau và không được đi lùi, chỉ duy nhất trên đoạn đường Điện Biên là các xe có thể quay đầu.

Bạn được ban tổ chức giao cho sắp xếp đoàn xe đi vào đường Chu Văn Thịnh theo đúng thứ tự, biết thứ tự các đoàn xe khi đi trên đường Trường Chinh. Hãy lập chương trình để làm việc đó.

**Dữ liệu vào:** Gồm nhiều bộ test, mỗi bộ test gồm dòng:

- Dòng đầu ghi số là số xe tham gia diễu hành;
- Dòng sau ghi n số nguyên dương là trật tự các xe trên đường Trường Chinh.

Kết thúc là số 0.

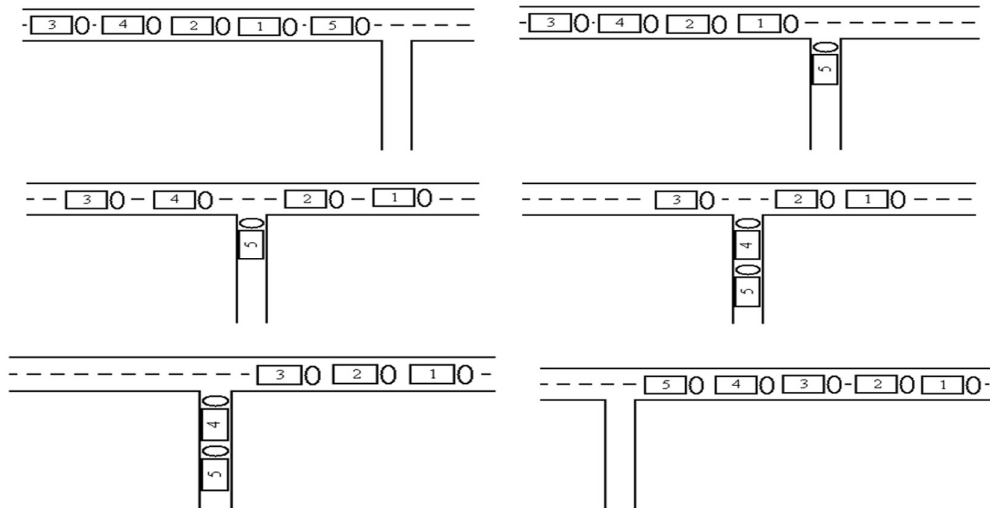
**Dữ liệu ra:** Ứng với mỗi bộ test, ghi ra **yes** nếu có thể xếp đúng được thứ tự, ghi ra **no** nếu không thể.

**Giới hạn:**  $1 \leq n \leq 10^5$ ; số test case không quá 10.

STPAR.inp	STPAR.out
5	yes
5 1 2 4 3	no
5	
4 3 5 1 2	
0	

Giải thích:

- Ta bố trí các xe (trong test ) tránh vào đường Điền Biên như hình dưới đây:



## Bài 7: 491 STRPN – Chuyển biểu thức toán học sang RPN

Bạn được cho một danh sách các biểu thức toán học đúng ở dạng trung tố chỉ chứa:

- Biến: là các chữ cái latin in thường a-z, (mỗi biến là một chữ cái)
- Các toán tử hai ngôi: +, -, \*, /, ^ (lũy thừa) với thứ tự ưu tiên như sau: +, - cùng độ ưu tiên thấp nhất; \*, / cùng độ ưu tiên thứ hai, ^ có độ ưu tiên cao nhất
- Các cặp dấu ngoặc

Hãy chuyển đổi biểu thức đó sang dạng hậu tố và giữ nguyên thứ tự các số hạng (RPN – ký pháp nghịch đảo Ba Lan)

**Dữ liệu vào:**

- Dòng đầu chứa số nguyên dương T là số biểu thức;
- T dòng tiếp theo, mỗi dòng chứa một biểu thức dạng trung tố.

**Dữ liệu ra:**

- Ứng với mỗi biểu thức dạng trung tố, in ra biểu thức RPN tương ứng trên một dòng.

**Giới hạn:**  $1 \leq T \leq 100$ ; Độ dài của các biểu thức trung tố không quá 400.

STRPN.inp	STRPN.out
3	abc*+
(a+(b*c))	ab+zx+*
((a+b)*(z+x))	at+bac++cd+^*
((a+t)*((b+(a+c))^(c+d)))	



## Bài 8: 492 STMEDIAN – Truy vấn trung vị

Bạn được cho một dãy  $n$  số nguyên dương không giảm và  $m$  truy vấn, với mỗi truy vấn bạn được yêu cầu in ra phần tử trung vị của dãy hiện tại và loại bỏ nó khỏi dãy (nếu dãy có  $n$  số theo thứ tự không giảm thì số trung vị là số thứ  $\lfloor \frac{n+1}{2} \rfloor$ ).

### Dữ liệu vào:

Dòng đầu chứa hai số nguyên dương  $n$  và  $m$ ;

Dòng thứ hai chứa  $n$  số nguyên dương  $a_1, a_2, \dots, a_n$  được liệt kê theo thứ tự không giảm.

### Dữ liệu ra:

Ghi ra trên một dòng kết quả truy vấn, hai số liên tiếp cách nhau một dấu cách.

Giới hạn:  $1 \leq m \leq n \leq 10^6$ ;  $1 \leq a_i \leq 10^9$

STMEDIAN.inp	STMEDIAN.out
5 4 1 2 3 3 5	3 2 3 1

## Bài 9: 493. STCONCERT – Hòa nhạc

Có  $N$  người đang xếp hàng chờ đợi để vào một buổi hòa nhạc. Mọi người cảm thấy buồn chán khi chờ đợi nên họ tìm những người thân của họ trong hàng đang xếp. Hai người A và B có thể nhìn thấy nhau nếu họ đang đứng ngay cạnh nhau hoặc không có người nào cao hơn A hoặc B đứng giữa. Hãy xác định số cặp có thể nhìn thấy nhau.

### Dữ liệu vào:

- Dòng đầu chứa số nguyên dương  $N$  là số người đang đứng xếp hàng.
- Dòng thứ hai chứa  $N$  số nguyên dương  $h_1, h_2, \dots, h_n$  theo thứ tự là độ cao của những người đang đứng trong hàng.

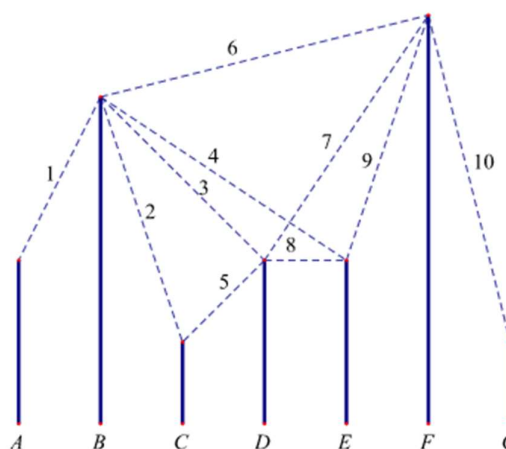
### Dữ liệu ra:

- Một số nguyên duy nhất là số cặp có thể nhìn thấy nhau.

Giới hạn:  $1 \leq n \leq 10^5$ ;  $1 \leq h_i \leq 10^9$ .

STCONCERT.inp	STCONCERT.out
7 2 4 1 2 2 5 1	10
<b>Giải thích:</b>	

- 10 cặp có thể nhìn thấy nhau là  $(A, B), (B, C), (B, D), (B, E), (C, D), (B, F), (D, F), (D, E), (E, F), (F, G)$  như hình vẽ:



## Bài 10. 496. STMASS – Khối lượng phân tử

Hợp chất hữu cơ chỉ gồm các nguyên tố C, H, O có khối lượng nguyên tử lần lượt là và .

Người ta viết công thức phân tử của hợp chất hữu cơ ở dạng "nén", ví dụ COOHHH là  $\text{CO}_2\text{H}_3$  hay  $\text{CH}(\text{CO}_2\text{H})(\text{CO}_2\text{H})(\text{CO}_2\text{H})$  là  $\text{CH}(\text{CO}_2\text{H})_3$ . Nếu ở dạng nén thì số lần lặp là một số  $2 \leq k \leq 9$ .

Cho công thức phân tử của một hợp chất hữu cơ, hãy tính khối lượng phân tử của nó.

### Dữ liệu vào:

Dòng đầu ghi số nguyên T là số bộ test;

T dòng tiếp theo, mỗi dòng chứa một xâu ký tự biểu diễn công thức phân tử của hợp chất hữu cơ.

### Dữ liệu ra:

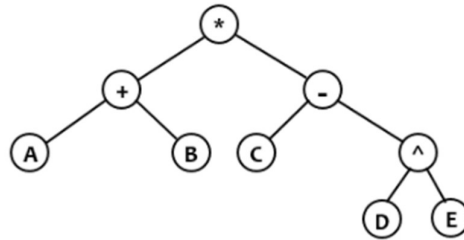
Gồm T dòng, mỗi dòng một số nguyên dương là khối lượng của hợp chất ở test tương ứng.

Giới hạn:  $1 \leq T \leq 1000$ ; độ dài công thức hóa học không quá 1000.

STMASS.inp	STMASS.out
3	45
COOH	148
$\text{CH}(\text{CO}_2\text{H})_3$	222
$((\text{CH})_2(\text{OH}_2\text{H})(\text{CHO}))_3$	

## Bài 11 499 RPNHEIGHT - Chiều cao cây RPN

Với mọi biểu thức số học gồm 5 phép toán hai ngôi  $+$ ,  $-$ ,  $*$ ,  $/$ ,  $^$  ở dạng hậu tố (RPN – ký pháp đảo ngược Ba Lan), ta có thể xây dựng một cây nhị phân tương ứng. Ví dụ với biểu thức hậu tố:  $AB+CDE^-*$  (mỗi toán hạng là một chữ cái Latinh) thì cây nhị phân tương ứng là như sau:



Yêu cầu: cho một biểu thức hậu tố, hãy tính chiều cao của cây nhị phân tương ứng với biểu thức đó.

Dữ liệu vào:

- Dòng đầu chứa số nguyên dương  $T$  là số biểu thức;
- $T$  dòng tiếp theo, mỗi dòng chứa một biểu thức RPN (các toán hạng là các chữ cái Latinh).

Dữ liệu ra:

- Ứng với mỗi biểu thức dạng RPN, in ra chiều cao cây nhị phân tương ứng trên một dòng.

Giới hạn:  $1 \leq T \leq 100$ ; độ dài các biểu thức không quá 5000.

RPNHEIGHT.inp	RPNHEIGHT.out
4	3
AB+CDE^-*	2
ABC*+	2
AB+ZX*+	4
AT+BAC++CD+^*	

## Bài 12. Tính giá trị biểu thức toán học.

Cho một chuỗi  $S$  là biểu thức toán học gồm các toán tử: cộng  $+$ , trừ  $-$ , nhân  $*$ , chia  $/$ , lũy thừa  $^$ ; các toán hạng gồm các số từ 1 đến 9 và các dấu ngoặc  $( )$ .

Ví dụ  $S = (2+3) * 5 + 4 + 6 * 8 - 2*3^2$ .

Chuỗi  $S$  có độ dài không quá 255 ký tự.

Tính giá trị của  $S$ .

BT.inp	BT.out
$2+3*5$	17
$(2+3)*5$	25
$2*3^2+5$	23