



Ngôn ngữ lập trình C++

§10§ BÀI TẬP LẬP TRÌNH NÂNG CAO



1☀. Xâu ngoặc đúng (1) – Brackets1.Cpp

Ta nói một dãy ngoặc đơn là dãy ngoặc đúng nếu dãy ngoặc đó có thể được tạo thành từ một biểu thức toán học đúng sau khi xóa đi các số hạng và các phép toán.

Ví dụ: Các dãy ngoặc đúng: (), ()(()), (()());

Các dãy ngoặc không đúng: ()(, ()(,))();

Yêu cầu: Cho dãy ngoặc đơn gồm các dấu ngoặc mở '(' và ngoặc đóng ')'. Kiểm tra xem dãy ngoặc đơn đã cho là dãy ngoặc đúng hay dãy ngoặc sai.

Dữ liệu cho trong file **Brackets1.Inp** gồm:

- Dòng đầu ghi số nguyên dương T ($T \leq 10$) là số dãy cần kiểm tra.
- T dòng sau, mỗi dòng ghi một dãy ngoặc đơn cần kiểm tra (độ dài không quá 10^4).

Kết quả ghi ra file **Brackets1.Out** gồm T dòng, mỗi dòng ghi 1 nếu dãy ngoặc tương ứng là dãy ngoặc đúng, ngược lại ghi 0.

Ví dụ:

Brackets1.Inp	Brackets1.Out
3	1
()()	0
()()	1
()	



2☀. Thay dấu ngoặc – Brackets2.Cpp

Ta nói một dãy ngoặc đơn là dãy ngoặc đúng nếu dãy ngoặc đó có thể được tạo thành từ một biểu thức toán học đúng sau khi xóa đi các số hạng và các phép toán.

Ví dụ: Các dãy ngoặc đúng: (), ()(()), (()());

Các dãy ngoặc không đúng: ()(, ()(,))();

Yêu cầu: Cho dãy gồm các dấu ngoặc mở '(' và ngoặc đóng ')' và dấu chấm '?'. Kiểm tra xem liệu có thể thay tất cả các dấu chấm '?' bởi dấu ngoặc mở hoặc dấu ngoặc đóng để nhận được dãy ngoặc đơn đúng hay không?

Dữ liệu cho trong file **Brackets2.Inp** gồm:

- Dòng đầu ghi số nguyên dương T ($T \leq 10$) là số dãy cần kiểm tra.
- T dòng sau, mỗi dòng ghi một dãy kí tự gồm kí tự ngoặc mở, ngoặc đóng, dấu chấm (độ dài không quá 10^4). Dữ liệu đảm bảo luôn có dấu chấm '?'.

Kết quả ghi ra file **Brackets2.Out** gồm T dòng, mỗi dòng ghi 1 nếu dãy kí tự tương ứng có thể thay thế để nhận được dãy ngoặc đúng, ngược lại ghi 0.

Ví dụ:



Brackets2.Inp	Brackets2.Out
3	1
(???)	0
(??)	1
(?	

**3. Cặp số hạng có ước chung lớn nhất**

Cho dãy số nguyên dương A_1, A_2, \dots, A_N . Tìm cặp số hạng A_i và A_j có ước chung lớn nhất $d = (A_i, A_j)$ có giá trị lớn nhất.

Dữ liệu cho trong file **MAXGCD.INP** gồm:

- Dòng đầu ghi số nguyên dương N là số các số hạng.
- Dòng thứ 2 ghi N số nguyên dương A_1, A_2, \dots, A_N .

Kết quả ghi ra file **MAXGCD.OUT** là giá trị lớn nhất của d đạt được.

Ví dụ:

MAXGCD.INP	MAXGCD.OUT
4	7
2 6 7 14	

Giới hạn:

- Có 50% số test ứng với $N \leq 1000$; $A_i \leq 10^9$;
- Có 50% số test ứng với $N \leq 100000$; $A_i \leq 10^6$.

**4. Xếp công việc (Hard Version 2)**

Có N công việc được đánh số thứ tự từ 1 đến N . Công việc thứ i có thời gian hoàn thành là A_i và hạn cuối cần hoàn thành là B_i . Nếu công việc thứ i được thực hiện xong trước hoặc ngay tại thời điểm B_i thì ta nói công việc i hoàn thành đúng thời hạn. Ngược lại công việc i không hoàn thành đúng thời hạn.

Yêu cầu: Hãy tìm cách sắp xếp các công việc sao cho số công việc thực hiện đúng hạn là nhiều nhất. Công việc đầu tiên có thể thực hiện từ thời điểm 0. Không được thực hiện nhiều hơn một công việc tại một thời điểm, thực hiện xong công việc này rồi mới chuyển sang công việc khác. Thời điểm hoàn thành công việc này và thời điểm bắt đầu công việc khác có thể bằng nhau. Công việc đầu tiên có thể bắt đầu từ thời điểm 0.

Dữ liệu cho trong file **WorkMax.Inp** gồm:

- Dòng 1 ghi số nguyên dương T ($T \leq 10$) là số bộ dữ liệu:
Mỗi bộ dữ liệu có dạng:
 - Dòng thứ nhất ghi số nguyên dương N là số công việc.
 - N dòng sau, dòng thứ i ghi hai số nguyên A_i và B_i .

Kết quả ghi ra file **WorkMax.Out** gồm T dòng. Mỗi dòng ghi số công việc nhiều nhất có thể thực hiện đúng hạn với bộ dữ liệu tương ứng.

Ví dụ:



WorkMax.Inp	WorkMax.Out
2	2
4	1
2 2	
3 3	
3 4	
8 10	
2	
2 2	
2 3	

Giới hạn:

- $3 \leq N \leq 10^5$;
- $1 \leq A_i \leq B_i \leq 10^9$.