

Universidade Federal de Goiás
Instituto de Informática

Disciplina: Algoritmos e Estrutura de Dados II

Professora: Nádia Félix

Avaliação 2 - Segunda chamada
Turma A

1. **(Valor 3.0)**

Pedrinho e Zezinho estão precisando estudar resolução de expressões matemáticas para uma prova que irão fazer. Para isso, eles querem resolver muitos exercícios antes da prova. Como sabem programar, então decidiram fazer um gerador de expressões matemáticas.

O gerador de expressões que eles criaram funciona em duas fases. Na primeira fase é gerada uma cadeia de caracteres que contém apenas os caracteres '{', '[', '(', '}', ']' e ')'. Na segunda fase, o gerador adiciona os números e operadores na estrutura criada na primeira fase. Uma cadeia de caracteres é dita bem definida (ou válida) se atende as seguintes propriedades:

- (a) Ela é uma cadeia de caracteres vazia (não contém nenhum caractere).
- (b) Ela é formada por uma cadeia bem definida envolvida por parênteses, colchetes ou chaves. Portanto, se a cadeia S é bem definida, então as cadeias (S) , $[S]$ e $\{S\}$ também são bem definidas.
- (c) Ela é formada pela concatenação de duas cadeias bem definidas. Logo, se as cadeias X e Y são bem definidas, a cadeia XY é bem definida.

Depois que Pedrinho e Zezinho geraram algumas expressões matemáticas, eles perceberam que havia algum erro na primeira fase do gerador. Algumas cadeias não eram bem definidas. Eles querem começar a resolver as expressões o mais rápido possível, e sabendo que você é um ótimo programador (e faz aula de estrutura de dados com a Professora Nádia ou o prof. Hebert) resolveram pedir que escreva um programa que dadas várias cadeias geradas na primeira fase, determine quais delas são bem definidas e quais não são.

Entrada

A entrada é composta por diversas instâncias. A primeira linha da entrada contém um inteiro T indicando o número de instâncias. Em seguida temos T linhas, cada uma com uma cadeia A.

Saída Para cada instância imprima uma linha contendo a letra S se a cadeia é bem definida, ou a letra N caso contrário.

Restrições

- $1 \leq T \leq 20$
- a cadeia de caracteres A tem entre 1 e 100000 caracteres.
- a cadeia de caracteres A contém apenas caracteres '{', '[', '(', '}', ']', ')', ',' e ' '.

Exemplo

Entrada:

```

12
()
[]
{}
()
} {
([ { } ] )
{ }( )[ ]
())
{[]}
(
((( [ { } { } ( [ [] ]) ({ }) {} $
((((((((({[[]}] ])))))))))

```

Saída

S
S
S
N
N

S
S
N
N
N
S
N

2. Os pacientes que chegam na fila do SUS passam por uma triagem imediatamente e vão para a fila de atendimento. Na triagem a enfermeira anota o horário de entrada do paciente e quantos minutos ele tem até que sua condição de saúde se torne crítica. Sabe-se que os pacientes são atendidos de 30 em 30 minutos (sempre nas horas cheias ou meias horas) quando na fila de atendimento. O início da triagem e do atendimento se dá às 7h da manhã, se não há nenhum paciente sendo atendido e a fila está vazia, o primeiro paciente é atendido no instante que chega na triagem. O médico atende até o último paciente na fila. A preocupação é se algum paciente atingiu uma condição crítica enquanto não tenha sido atendido. Para tanto você foi convidado para verificar na fila quantos pacientes atingem a condição crítica.

Entrada

A entrada começa com uma linha com o número inteiro N , $0 < N < 25$; o número de pacientes que chegam à triagem. A seguir são N linhas com os valores inteiros H , M e C , com $7 < H < 19$, e $0 \leq M < 60$, a hora e minuto que o paciente chega à triagem. O paciente da linha i sempre chega antes que, e no máximo junto com, o paciente da linha $i + 1$. E $0 \leq C \leq 720$ o número de minutos antes do paciente atingir a condição crítica de saúde.

Saída

Imprima o número de pacientes que atingiram a condição crítica ainda na fila de atendimento.

Exemplo

Entrada:

4
7 0 20

```
7 0 30
7 30 20
8 15 30
```

Saída

1

Entrada

5

10 20 50

10 30 30

11 10 20

12 0 0

12 10 30

Saída

0