

# Drzewo BST w wersji nieco mniej light

W tym zadaniu należy wykonać symulację drzewa BST - (Binary Search Tree).

Wykonujemy następujące czynności:

- i - wstaw element **val** do drzewa
- s - wyszukaj element **val** podając kolejno odwiedzane elementy drzewa BST i jeśli go znajdzie, wypisz jego poprzednik i następnik w formacie: następnik: liczba poprzednik: liczba, natomiast jeśli go nie znajdzie wypisz dodatkowo komunikat: element nie występuje w drzewie
- m - wyszukaj minimum
- M - wyszukaj maksimum

## Wejście

W pierwszym wierszu jedna liczba **t** nie większa niż milion, określająca liczbę zapytań.

W kolejnych **t** wierszach instrukcje postaci:

- i [wartość]
- s [wartość]
- m
- M

Wartości są unikatowe i mieszczą się w typie int.

## Wyjście

Dla operacji **s** wypisz kolejne odwiedzane wierzchołki drzewa, dla **m** wypisz najmniejszą wartość, dla **M** największą wartość.

## Przykład

Wejście:

```
20
i 12
s 17
s 12
m
i 16
i 4
M
s 5
i 14
i 0
s 7
m
M
M
i 18
M
```

s 0  
s 12  
s 0  
M

**Output:**

12  
element nie wystepuje w drzewie  
12  
poprzednik: brak  
nastepnik: brak  
12  
16  
12 4  
element nie wystepuje w drzewie  
12 4  
element nie wystepuje w drzewie  
0  
16  
16  
18  
12 4 0  
poprzednik: brak  
nastepnik: 4  
12  
poprzednik: 4  
nastepnik: 14  
12 4 0  
poprzednik: brak  
nastepnik: 4  
18