

子題 2：後序運算式(postfix)。(程式執行限制時間: 2 秒) 15 分

求一個運算式值，一般在電腦中，常用的運算式表示法有下列三種表示法：

(1)中序運算式(infix) :<運算元> <運算子> <運算元>，如 $A+B$ 。

(2)前序運算式(prefix) :<運算子> <運算元> <運算元>，如 $+AB$ 。

(3)後序運算式(postfix) :<運算元> <運算元> <運算子>，如 $AB+$ 。

後序運算式(postfix)有別於我們習慣的運算式寫法(中序運算式,infix)，是把運算子寫在運算元之後，雖然對人來說可讀性很低，可是對電腦是很方便的。後序運算式的好處還可以不用考慮中序運算式的先乘除後加減的問題。

比如我們習慣 $3 + 5$ 這樣的式子，就是中序運算式，同一個式子改成後序寫法即為 $3 5 +$

輸入後序運算式，求出運算式值。

後序表示法 $9 2 3 5 + * +$ 對應到中序運算式 $9 + 2 * (3 + 5)$

輸出: 25

提示： 後序運算式的演算法

(1)從後序運算式字串中取出一<運算元>或<運算子>，分析之。

(2)若為運算元，則將該運算元放入(PUSH)運算元堆疊。

若為運算子，則由運算元堆疊中連續彈出(POP)兩個運算元，先彈出者(POP)為 B，後彈出者(POP)為 A，再依 “A 運算子 B”之運算式求得其值，運算式結果放入(PUSH)運算元堆疊。

(3)回到(1)，直到後序運算式字串完全處理完畢。

彈出(POP)運算元堆疊內之值，即為答案。

輸入說明：

第一列的數字 n 代表有幾筆資料要測試， $2 \leq n \leq 20$ ，第二列起為每組的測試資料，之後每一列為每筆的測試資料。輸入一個後序運算式的字串包含數個運算元及運算子，為了方便讀取，所有的運算元與運算子之間以一個或多個空白隔開。運算子包含 $+$ $-$ $*$ $/$ 等四個，運算元則為 $0 \sim 2^{31} - 1$ 的整數。

輸出說明：

每組測試資料輸出該後序運算式的結果。

為避免小數誤差問題，運算結果必定為整數，運算過程中不會出現小數的結果，不會有 $4/3$ 或 $3/2$ 無法整除的測試資料，因此請放心使用整數來進行運算。也不會有溢位運算。

輸入檔案 1：【檔名：in1.txt】

```
3
3 5 +
9 2 3 5 + * +
6 3 / 1 4 - * 3 + 8 -
```

輸入檔案 2：【檔名：in2.txt】

```
3
2 3 + 4 *
2 3 4 + * 5 *
2 123 -
```

輸出範例：【檔名：out1.txt】

```
8
25
-11
```

輸出範例：【檔名：out2.txt】

```
20
70
-121
```