

穴埋め初級問 3 逆ポーランド記法 (2003 年春 FE 午後問 4 改)

問 次のプログラムの説明及びプログラムを読んで、設問に答えよ。

[プログラムの説明]

逆ポーランド表記法で表された数式を、スタックを用いて計算する副プログラム Calc である。

- (1) 逆ポーランド表記法で表された数式は、文字型の 1 次元配列 Ex の各要素 Ex[0], ..., Ex[Lp] ($Lp \geq 3$) に 1 文字ずつ格納されている。
- (2) 数式は、正負の整数及び一つ以上の四則演算記号からなる。ただし、正の整数の場合はプラス記号を付けない。
- (3) 先頭の整数を除き、整数の前には一つの空白文字を置く。
- (4) 計算は実数で行う。計算の途中で次の状態になった場合は、副プログラム Abort() を呼び出してプログラムを終了する。
 - ① ゼロ除算が行われた。
 - ② スタックの範囲外を参照した。
- (5) Calc は、実数をスタックに積む副プログラム Push と、スタックから実数を取り出す副プログラム Pop を使用する。各副プログラムの引数の仕様を表 1 ~ 3 に示す。また、数字 1 文字を実数に変換する関数 ToReal も使用する。
- (6) 実数型の 1 次元配列 Stack, Stack の最大要素番号を示す定数 MAX, スタックの操作位置を示す変数 Sp が、大域変数として定義されている。Sp の初期値は 0 である。
- (7) 逆ポーランド表記法で表された数式は、正しいものとする。

例： 中置表記法による数式

逆ポーランド表記法による入力

		0	1	2	3	4	5	6	7	8	9	
(1 + 2) * (3 - 4)	Ex	1	△	2	+	△	3	△	4	-	*	Lp = 9
		0	1	2	3	4	5	6	7			
12 / (- 345)	Ex	1	2	△	-	3	4	5	/			Lp = 7

注 △は空白文字を示す

表 1 Calc の引数の仕様

変数名	入力/出力	意 味
Ex[]	入力	逆ポーランド表記法の数式が格納された文字型の 1 次元配列
Lp	入力	文字型の 1 次元配列 Ex の最後の要素番号 ($Lp \geq 3$)
Ret	出力	計算結果

表 2 Push の引数の仕様

変数名	入力/出力	意 味
T	入力	スタックに積む実数

表3 Pop の引数の仕様

変数名	入力/出力	意味
T	出力	スタックから取り出す実数

[プログラム]

○副プログラム名: Calc(Ex[], Lp, Ret)

○文字型: Ex[]

○整数型: Lp, Cp

○実数型: Ret, X, Y, T

○論理型: NumF, NegF

・Cp ← 0

・T ← 0

・NumF ← false

・NegF ← false

■ Cp ≤ Lp

▲ Ex[Cp] = 数字

・T ← T × 10 + ToReal(Ex[Cp])

{ToReal() は数字 1 文字を実数型に変換する関数}

・NumF ← true

▲ a

▲ NegF = true

・T ← -T

・NegF ← false

▼

・Push(T)

▼ b

▼

・NumF ← false

▲ Ex[Cp] = "+"

▼

・Pop(Y)

・Pop(X)

・T ← X + Y

・Push(T)

▼

・T ← 0

▲ Ex[Cp] = "-"

▲ Cp ≠ Lp AND Ex[Cp+1] = 数字

▼ c

▼

・Pop(Y)

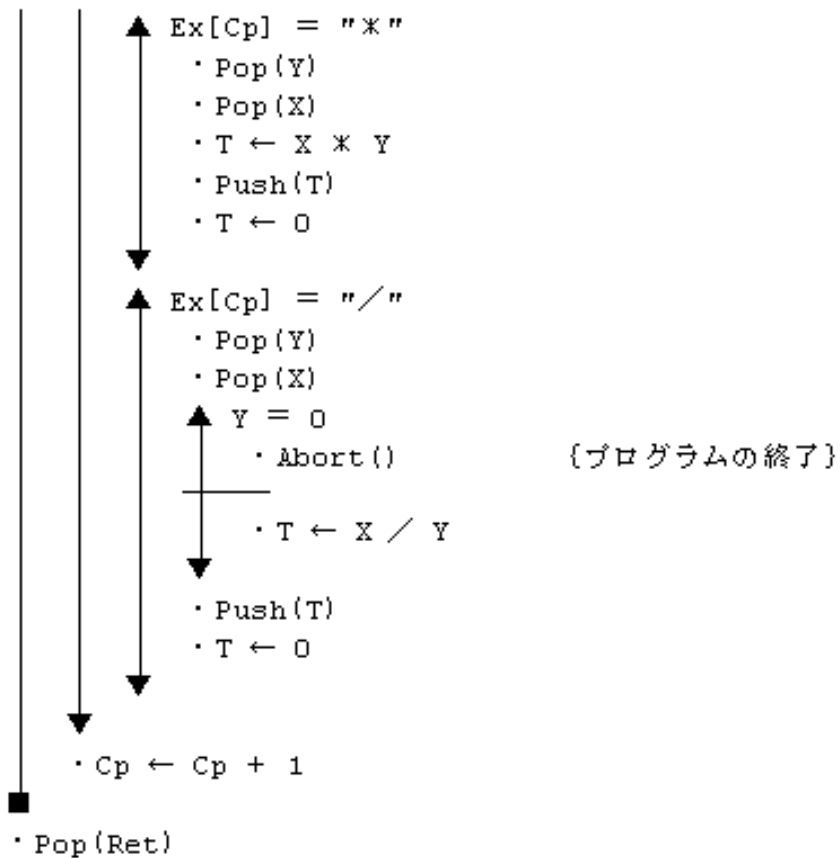
・Pop(X)

・T ← X - Y

・Push(T)

▼

・T ← 0



- 副プログラム名: Push(T)
- 実数型: T
- 外部参照: Stack[], Sp



- 副プログラム名: Pop(T)
- 実数型: T
- 外部参照: Stack[], Sp



設問 プログラム中の に入れる正しい答えを記述せよ。