

Una `CalcolatricePolacca` realizza una calcolatrice, sfruttando la *notazione polacca postfissa* (detta anche *notazione polacca inversa*). In tale notazione, gli operatori sono posti dopo i loro operandi. Nella notazione infissa tradizionale, le parentesi sono usate per indicare l'ordine delle operazioni. Tuttavia, nella notazione polacca postfissa, il posizionamento degli operatori elimina la necessità di parentesi, semplificando le espressioni e riducendo l'ambiguità. L'idea di base è che prima bisogna inserire in una pila, immaginata in verticale, gli operandi, per poi effettuare le varie operazioni (unarie o binarie che siano). Nel caso di operazione unaria, questa opererà sull'operando che si trova in cima alla pila, calcolerà il risultato, e il risultato andrà a rimpiazzare l'operando in cima alla pila (quello stesso su cui ha operato). Nel caso di operazioni binarie, essa lavorerà sui due operandi che trova in cima alla pila e calcolerà il risultato. Il risultato rimpiazzerà i due operandi utilizzati. La pila **deve essere illimitata**. Si assuma di dover effettuare operazioni **solo tra numeri interi**. **Nota Bene: l'uso di un vettore, seppur dinamico, comporta una penalizzazione di 12 punti.**



Implementare le seguenti operazioni per una `CalcolatricePolacca`.

--- PRIMA PARTE ---

✓ `CalcolatricePolacca c;`

Costruttore di default che inizializza una `CalcolatricePolacca` senza alcun operando.

✓ `cout << c;`

Operatore di uscita per il tipo `CalcolatricePolacca`. L'uscita ha la forma seguente:

```
#####          ← la prima e l'ultima riga è composta da 8 cancelletti
Op1: 3          ← questo è l'operando che è stato inserito per ultimo
Op2: 5
Op3: 4          ← questo è l'operando che è stato inserito per primo
#####
```

L'operando inserito più di recente deve apparire in cima alla pila, ed avere associata la stringa "Op1: " (notare la presenza **di un singolo spazio** dopo i due punti). Gli operandi che si trovano sotto, avranno numerazione progressiva.

Se a questo punto venisse invocata l'operazione di moltiplicazione (descritta sotto), il nuovo stato risulterebbe essere il seguente:

```
#####
Op1: 15         ← questo è il prodotto, che ha preso il posto degli operandi su cui ha operato
Op2: 4          ← questo è l'operando che è stato inserito per primo
#####
```

Infine si noti che una `Calcolatrice Polacca` vuota verrebbe visualizzata nel seguente modo:

```
#####
#####
```

✓ `c.ins(v);`

Operazione che inserisce l'intero `v` in cima alla pila. L'operazione non può fallire, perché si assume una pila di dimensione illimitata (ossia memoria RAM infinita).

✓ `c.moltiplica();`

Operazione che moltiplica i due operandi in cima alla pila e sostituisce ad essi il loro prodotto. Nel caso la `CalcolatricePolacca c` non contenga almeno due operandi, l'operazione non deve fare nulla (e pertanto `c` deve rimanere invariata).

✓ `c.somma();`

Operazione che somma i due operandi in cima alla pila e sostituisce ad essi la loro somma. Nel caso la `CalcolatricePolacca c` non contenga almeno due operandi, l'operazione non deve fare nulla (e pertanto `c` deve rimanere invariata).

--- SECONDA PARTE ---

✓ Distruttore

Solo se necessario, implementare il distruttore per il tipo `CalcolatricePolacca`.

✓ `c.duplica();`

Operazione che duplica l'operando che si trova in cima alla pila, ossia aggiunge in cima alla pila un nuovo operando, con valore coincidente a quello che precedentemente si trovava in cima alla pila. A seguito di questa operazione avremo in cima alla pila due operandi con ugual valore. Nel caso la `CalcolatricePolacca c` sia vuota, l'operazione non deve fare nulla (ossia `c` deve rimanere invariata).

✓ `c.opposto();`

Operazione che cambia il segno all'operando che si trova in cima alla pila. Nel caso la `CalcolatricePolacca c` sia vuota, l'operazione non deve fare nulla (ossia `c` deve rimanere invariata).

✓ `c3 = c2;`

Operazione di assegnamento, che rende la `CalcolatricePolacca c3` identica alla `CalcolatricePolacca c2`.

✓ `c.rimuoviNegativi();`

Operazione che rimuove tutti gli operandi negativi eventualmente presenti nella `CalcolatricePolacca c` (inoltre **non deve** modificare l'ordine di quelli positivi o nulli).

Esempio, qualora questa fosse la situazione PRIMA della `rimuoviNegativi()`

```
#####  
Op1: 88  
Op2: -23  
Op3: 13  
Op4: -19  
Op5: -49  
#####
```

Questa sarebbe la situazione DOPO la `rimuoviNegativi()`

```
#####  
Op1: 88  
Op2: 13  
#####
```

Mediante il linguaggio C++, realizzare il tipo di dato astratto definito dalle precedenti specifiche. Non è permesso utilizzare funzionalità della libreria STL come il tipo `string`, il tipo `vector`, il tipo `list`, ecc.

USCITA ATTESA

--- PRIMA PARTE ---

```
#####  
Op1: 3  
Op2: 5  
#####
```

```
#####  
Op1: 15  
#####
```

```
#####  
Op1: 4  
Op2: 15  
#####
```

```
#####  
Op1: 19  
#####
```

```
#####  
Op1: 3  
Op2: 5  
Op3: 4  
Op4: 19  
#####
```

```
#####  
Op1: 15  
Op2: 4  
Op3: 19  
#####
```

```
#####  
Op1: 19  
Op2: 19  
#####
```

--- SECONDA PARTE ---

```
Test della funzione 'duplica' e della funzione 'opposto' (deve mostrare 'Op1: -49')  
#####  
Op1: -49  
#####
```

```
Test operatore di assegnamento  
Contenuto di c3 prima dell'ass. (deve mostrare gli operandi: 'Op1: 22' e 'Op2: 11')  
#####  
Op1: 22  
Op2: 11  
#####
```

```
Contenuto di c3 dopo avergli assegnato c2 (ora dovra' contenere solo 'Op1: -49')  
#####  
Op1: -49  
#####
```

```
Test della funzione 'rimuoviNegativi'  
Situazione PRIMA della 'rimuoviNegativi'  
#####  
Op1: 88  
Op2: -23  
Op3: 13  
Op4: -19  
Op5: -49  
#####
```

```
Situazione DOPO la 'rimuoviNegativi'  
#####  
Op1: 88  
Op2: 13  
#####
```

Note per la consegna:

Affinché l'elaborato venga considerato valido, il programma **deve** riprodurre, in maniera esatta, almeno la prima parte dell'output atteso. In questo caso, i docenti procederanno alla valutazione dell'elaborato **solo se** lo studente avrà completato l'autocorrezione del proprio elaborato. In tutti gli altri casi (per esempio: il programma non compila, non collega, non esegue o la prima parte dell'output non coincide con quella attesa), l'elaborato è considerato **insufficiente** e, pertanto, **non verrà corretto** da parte dei docenti.