

Un `NastroTrasportatore` è un dispositivo che trasporta pezzi meccanici. I pezzi vengono inseriti da un'estremità del nastro e vengono rimossi dall'altra estremità. Ogni pezzo è caratterizzato da un identificatore di lunghezza variabile. In particolare, l'identificatore è composto da una lettera maiuscola seguita da una o più cifre. Mentre i pezzi circolano sul nastro, un operatore effettua il controllo della qualità su pezzi scelti a caso dal nastro, marcandoli come controllati.

Implementare le seguenti operazioni che possono essere fatte su `NastroTrasportatore`:

--- **PRIMA PARTE** --- *(qualora siano presenti errori di compilazione, collegamento o esecuzione in questa parte, l'intera prova sarà considerata insufficiente e pertanto non sarà corretta)*

✓ **`NastroTrasportatore n;`**

Costruttore che inizializza un nastro trasportatore. Inizialmente, non ci sono pezzi sul nastro.

✓ **`n.aggiungi(id);`**

Operazione che inserisce sul nastro trasportatore il pezzo avente identificatore `id`. L'operazione può fallire se l'identificatore del pezzo non ha il formato corretto.

✓ **`n.rimuovi();`**

Operazione che rimuove dal nastro trasportatore il pezzo che ha raggiunto la fine del nastro.

✓ **`cout << s;`**

Operatore di uscita per il tipo `NastroTrasportatore`. L'uscita ha il seguente formato:

`(C878) = (A1625) = (B42)`

L'output mostrato a video corrisponde a un nastro su cui sono presenti tre pezzi. Il nastro si muove da **sinistra a destra**, quindi `B42` è il primo pezzo a essere stato inserito e `C878` l'ultimo. Nel caso non ci siano pezzi sul nastro, l'output mostrato è il seguente:

`<vuoto>`

✓ **`NastroTrasportatore n1(n);`**

Costruttore di copia per il tipo `NastroTrasportatore`.

--- **SECONDA PARTE** ---

✓ **`n.controlla(i);`**

Operazione che effettua il controllo della qualità sull'`i`-esimo pezzo che si trova sul nastro. Il primo pezzo sul nastro (partendo da sinistra) ha posizione `i=1`. In particolare, il pezzo viene tolto momentaneamente dal nastro, viene marcato come controllato e poi ricompare sul nastro nella posizione immediatamente precedente.

Esempio: dato il seguente nastro trasportatore:

`(C878) = (A1625) = (B42)`

Invocando la funzione con `i=2`, il nastro si modifica nel seguente modo:

`(A1625*) = (C878) = (B42)`

dove il pezzo avente identificatore `A1625` è stato marcato come controllato (nell'output deve comparire il carattere `*`) e reinserto dopo il pezzo che lo seguiva sul nastro, ovvero quello avente identificatore `C878`.

Se il pezzo è già marcato come controllato, il controllo non deve essere rieseguito e il nastro rimane inalterato.

✓ **`int(n);`**

Operatore di conversione, che consente di utilizzare un `NastroTrasportatore` ovunque serva un intero. L'operatore ritorna il numero di pezzi presenti sul nastro sui quali è stato effettuato il controllo.

✓ ~NastroTrasportatore ();

Distruttore.

Mediante il Linguaggio C++, realizzare il tipo di dato astratto **NastroTrasportatore**, definito dalle precedenti specifiche. **Gestire le eventuali situazioni di errore.**

## USCITA CHE DEVE PRODURRE IL PROGRAMMA

--- PRIMA PARTE ---

Test del costruttore

<vuoto>

Test della aggiungi()

(F024)

(B42)=(F024)

(A1625)=(B42)=(F024)

(C878)=(A1625)=(B42)=(F024)

Test della rimuovi()

(C878)=(A1625)=(B42)

Test del costruttore di copia

(C878)=(A1625)=(B42)

--- SECONDA PARTE ---

Test della controlla()

(A1625\*)=(C878)=(B42)

(A1625\*)=(B42\*)=(C878)

Test di int()

2

Test del distruttore

(n2 e' stato distrutto)