



✓ `p.aggiungi(col, pos);`

Funzione membro che aggiunge al `Portafoto p` una foto lungo la colonna `col` alla posizione `pos`.

Si noti che:

- Le foto sono aggiunte consecutivamente lungo una colonna, ovvero una foto può essere collocata in posizione 2 solo se la posizione 1 della medesima colonna è occupata.
- Qualora la posizione della nuova foto sia occupata, l'occupante e tutte le successive scrono di una posizione verso il basso così da liberarla. L'associazione tra le foto deve essere aggiornata coerentemente.
- Qualora la posizione della nuova foto non sia ammissibile la funzione fallisce.

La funzione restituisce `false` in caso di fallimento, `true` altrimenti.

✓ `p.associa(col, pos, col_ass, pos_ass);`

Funzione che aggiorna la foto in posizione `pos` lungo la colonna `col` associandole la foto in posizione `pos_ass` lungo la colonna `col_ass`. Se la foto era già associata ad un'altra, questa informazione viene sovrascritta e dunque persa. Se una delle due foto non è presente nel `Portafoto p`, la funzione restituisce `false` lasciando inalterata la struttura dati, altrimenti `true`.

--- **Metodi invocati nella SECONDA PARTE di main.cpp:** ---

✓ `p.elimina(col, pos);`

Funzione membro che elimina una foto in posizione `pos` lungo la colonna `col` dal `Portafoto p` e compatta la colonna facendo scorrere le eventuali foto successive verso l'alto di una posizione. Se una più foto sono associate a quella da eliminare, le prime devono essere modificate in modo da risultare non più associate ad alcuna foto. La funzione restituisce `true` in caso di corretta eliminazione, altrimenti `false`.

✓ `p=q;`

Operatore di assegnamento per `Portafoto`.

✓ `~Portafoto();`

Distruttore.

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

**SUGGERIMENTO 1:** Può tornare molto comodo implementare una funzione di utilità `get_foto(col, pos)` che restituisca un riferimento alla foto in posizione `pos` lungo la colonna `col`.

**SUGGERIMENTO 2:** Ragionare bene su come rappresentare l'associazione tra foto all'interno della struttura dati. Una buona scelta può facilitare molto l'implementazione, specialmente considerando le funzioni `aggiungi` ed `elimina`.

---

USCITA CHE DEVE PRODURRE IL PROGRAMMA

--- PRIMA PARTE ---

Test del costruttore

Associazioni:

Test della aggiungi

1  
0

---

```
| | |
| | |
* * *
| |
| |
* *
|
|
*
```

Associazioni:

Test della associa

1  
0

---

```
| | |
| | |
* * *
| |
| |
* *
|
|
*
```

Associazioni:

(1,1) -> (3,1)  
(2,2) -> (1,2)  
(2,3) -> (1,1)

Test dell'aggiornamento delle associazioni dopo aggiunta

---

```
| | |
| | |
* * *
| | |
| | |
* * *
| |
| |
* *
```

Associazioni:

(1,1) -> (3,2)  
(2,2) -> (1,2)  
(2,3) -> (1,1)

--- SECONDA PARTE ---

Test della elimina

1  
0

```
-----  
| | |  
| | |  
* * *  
| | |  
| | |  
* * *  
|  
|  
*
```

Associazioni:

(1,1) -> (3,2)  
(2,2) -> (1,2)  
(2,3) -> (1,1)

Test dell'aggiornamento delle associazioni dopo eliminazioni

```
-----  
| | |  
| | |  
* * *  
| |  
| |  
* *
```

Associazioni:

(1,1) -> (3,2)  
(2,2) -> (1,1)

Operatore di assegnamento

```
-----  
| | | |  
| | | |  
* | * *  
| |  
| |  
* *  
|  
|  
*  
|  
|  
*
```

Associazioni:

(4,1) -> (1,1)  
(4,2) -> (5,2)  
(5,3) -> (5,2)

Test del distruttore

Oggetto distrutto

---

**Nota finale.** Affinché l'elaborato venga considerato valido, il programma deve produrre 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 (sia della prima che della seconda parte).

In **tutti** gli altri casi (il programma non compila, non collega, non esegue, la prima parte dell'output non coincide con quella attesa o lo studente non effettua l'autocorrezione), l'elaborato è considerato **insufficiente e non verrà corretto**.