

Un `Mosaico` consiste in una matrice di caselle quadrate tutte della stessa dimensione. Le caselle del `Mosaico` possono essere vuote o occupate da una tessera colorata. I possibili colori delle tessere sono `GIALLO`, `BIANCO`, `ARANCIONE` e `VERDE`. Le righe e le colonne della matrice sono numerate a partire da 1. Implementare le seguenti operazioni che possono essere effettuate su un `Mosaico`:

- `Inizializza(m, r, c);`
Funzione che inizializza un mosaico `m`, con un numero `r` di righe e un numero `c` di colonne. Inizialmente il mosaico non contiene tessere.
- `Inserisci(m, i, j, col);`
Funzione che inserisce nel mosaico `m` una nuova tessera di colore `col`, nella posizione: riga `i` e colonna `j`. Se la casella è occupata, la funzione lascia il mosaico inalterato.
- `MaxVuote(m);`
Funzione che restituisce il massimo numero di caselle consecutive vuote, tutte sulla stessa riga della matrice.
- `Diagonale(m);`
Funzione che restituisce `true` se tutte le tessere sulla diagonale principale e secondaria del mosaico `m` sono di colore `GIALLO`. La funzione restituisce `false` altrimenti.
- `CreaLista(m);`
Funzione che crea e restituisce una lista con un elemento per ogni tessera, contenente il suo colore, del mosaico `m`. Le tessere appaiono nella lista per righe: la lista conterrà prima tutte le tessere colorate della prima riga, poi tutte quelle colorate della seconda e così via. Il primo elemento della lista, quindi, corrisponde alla prima tessera della prima riga; l'ultimo elemento della lista corrisponde all'ultima tessera dell'ultima riga.
- `Stampa(m);`
Funzione di uscita di un `Mosaico`. L'uscita ha il seguente formato:

G-AAAV
BB-VVB

Per ogni riga viene mostrato G per una tessera di colore `GIALLO`, B per `BIANCO`, A per `ARANCIONE` e V per `VERDE`. Se la casella è vuota viene mostrato il carattere -. Nell'esempio il mosaico ha 2 righe e 6 colonne. Nella prima riga vi è nell'ordine: una tessera di colore `GIALLO`, una casella vuota, tre tessere di colore `ARANCIONE` ed infine una tessera di colore `VERDE`. Non ci sono spazi tra i caratteri, ma ogni riga della matrice deve essere stampata su una riga separata.

Mediante il linguaggio C++, realizzare il tipo `Mosaico` definito dalle precedenti specifiche utilizzando le strutture. Gestire le eventuali situazioni di errore.

Esempio di funzione main()

Contenuto dello stream d'output standard mostrato nel commento */* */*

```
int main(){
    Mosaico m;
    Inizializza (m, 2, 6);
    Stampa(m);

    /*      -----
           -----
    */

    Inserisci(m, 2,3, BIANCO); // eventualmente sostituire BIANCO con codifica scelta
    Inserisci(m, 1, 5, GIALLO); // eventualmente sostituire GIALLO con codifica scelta
    Stampa(m);

    /*      ----G-
           --B---
    */

    int k= MaxVuote(m);
    cout <<<k << endl;

    /* 4 */

    bool b = Diagonale (m);
    if (b)
        cout << "true" << endl;
    else
        cout << "false" << endl;

    /* false */

    elem* p = CreaLista(m);    \\ elem è il nome della struttura dell'elemento della lista

    return 0;

}
```