

Canvas è un programma di grafica che permette di disegnare figure su un foglio virtuale. Il foglio è visto come una griglia di caselle di dimensione $m \times n$. La casella in alto a sinistra ha coordinate $(0,0)$. Le figure vengono disegnate colorando le caselle del foglio. I colori consentiti sono bianco, nero e grigio. Realizzare le seguenti operazioni che possono essere effettuate con Canvas:

✓ **Canvas c(m,n)**

Costruttore, che crea un foglio virtuale di dimensione $m \times n$. Inizialmente, il foglio ha colore bianco.

✓ **c.clear(i,j)**

Operazione che imposta a bianco il colore della casella (i,j) del foglio virtuale c .

✓ **c.darken(i,j)**

Operazione che colora le caselle sulle righe $\geq i$ e colonne $\geq j$ del foglio virtuale c nel seguente modo: se la casella era bianca diventa grigia, se era grigia diventa nera, se era nera resta nera.

✓ **c1 = c**

Operatore di assegnamento, che sostituisce il valore del foglio $c1$ con quello del foglio c .

✓ **cout << c**

Operatore di uscita per il tipo Canvas. L'uscita visualizza il contenuto del foglio virtuale. Le caselle di colore bianco sono identificate dal punto ('.'), quelle di colore grigio dal carattere 'G', quelle di colore nero dal carattere 'N'. L'output seguente corrisponde a una figura disegnata su un foglio virtuale di dimensione 3×4 , dove la prima colonna di caselle è di colore grigio, le ultime tre caselle dell'ultima riga hanno colore nero, tutte le altre caselle hanno colore bianco:

```
G . . .
G . . .
G N N N
```

✓ **c%r**

Operatore modulo, che ruota la figura c di un numero di gradi pari ad $r \cdot 180^\circ$.

✓ **~Canvas()**

Distruttore.

Utilizzando il linguaggio C++, realizzare il tipo di dati astratto Canvas definito dalle precedenti specifiche. Individuare eventuali situazioni di errore, e metterne in opera un corretto trattamento. Si faccia riferimento al main riportato a pagina 2.

```

// file main.cpp
#include "compito.h"

int main(){
    //test costruttore
    Canvas c(3,4);
    Canvas c1(1,1);

    //test operator<<
    cout << "Canvas c" << endl;
    cout << c << endl;

    cout << "Canvas c1" << endl;
    cout << c1 << endl;

    //test operatore di assegnamento
    c1 = c;
    cout << c1 << endl;

    //test darken
    c.darken(1,1);
    cout << c << endl;
    c.darken(2,2);
    cout << c << endl;

    //test clear
    c.clear(1,1);
    cout << c << endl;

    //test operator%
    c%1;
    cout << c << endl;

    {
        // test costruttore di copia
        Canvas c2(c);
        cout << c2 << endl;

    } // test distruttore (c2 sta per essere distrutto)
    return 0;
}

```

Uscita attesa

Canvas c

```

. . . .
. . . .
. . . .

```

Canvas c1

```

.

```

```

. . . .
. . . .
. . . .

```

```

. . . .
. G G G
. G G G

```

```

. . . .
. G G G
. G N N

```

```

. . . .
. . G G
. G N N

```

```

N N G .
G G . .
. . . .

```

```

N N G .
G G . .
. . . .

```