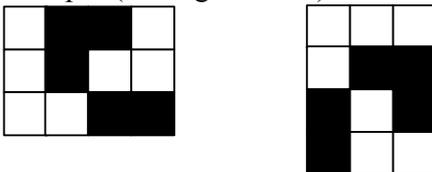


Un'immagine in bianco e nero viene rappresentata in un computer come un insieme di piccoli quadratini, chiamati pixel, che possono assumere o valore 0 (corrispondente al bianco) oppure valore 1 (corrispondente al nero). La dimensione dell'immagine viene data come $n \times m$, dove n e m rappresentano le dimensioni della larghezza e dell'altezza dell'immagine. Sia n che m sono espresse in numero di pixel. Per esempio, un'immagine 100×200 è un'immagine larga 100 pixel ed alta 200.

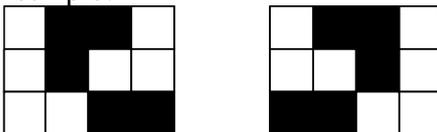
Definire le operazioni seguenti che possono essere effettuate su un'immagine:

- **Immagine i(n,m)**
Costruttore che crea un'immagine di dimensione $n \times m$ completamente bianca.
- **~Immagine()**
Distruttore.
- **cout << i**
L'operatore di uscita per il tipo **Immagine**. Per stampare i pixel, si stampi un carattere corrispondente al codice ASCII 219 per il bianco e 0 per il nero.
- **i.set(x,y)**
Operazione che inizializza al colore nero il pixel di coordinate (x,y). La funzione restituisce **false** se le coordinate (x,y) non sono comprese nell'immagine.
- **i.conta(x,y)**
Operazione che restituisce il numero di pixel bianchi vicini al pixel di coordinate (x,y). Un pixel è vicino al pixel di coordinate (x,y) se ha un lato in comune. La funzione restituisce **-1** se le coordinate (x,y) non sono comprese nell'immagine.
- **a - b**
Operatore di sottrazione tra due immagini che restituisce l'intersezione delle due immagini. Si tenga presente che le dimensioni delle due immagini possono essere diverse. L'intersezione viene realizzata applicando l'AND tra pixel corrispondenti.
- **i.ruota()**
Operazione che ruota l'immagine di 90 gradi in senso orario.
Esempio (immagine 4×3): a sinistra l'immagine originale ed a destra l'immagine ruotata



- **i.capovolgi()**
Operazione che capovolge l'immagine: a sinistra l'immagine originale ed a destra l'immagine capovolta

Esempio:



Utilizzando il linguaggio C++, realizzare il tipo di dati astratti Immagine, definito dalle precedenti specifiche.

NOTE SULLO SVOLGIMENTO DELLA PROVA PRATICA:

- Effettuare il **login**
Nome: studenti
Password: studenti
- Aprire il *Dev-C++* (dal Menù *Avvio* (o *Start*) nella barra degli strumenti in fondo allo schermo, selezionare *Programmi* e quindi *Dev-C++*)
- **Prima di iniziare a svolgere l'elaborato**, selezionare la voce *Identifica studente* nel menù *Strumenti* all'interno dell'ambiente *Dev-C++* e inserire i dati richiesti
- Dal menu *File* del *Dev-C++*, aprire il progetto *esainf.dev* presente nella cartella *c:\esame\esaInf*. Il progetto contiene tre file, denominati *compito.h*, *compito.cpp* e *main.cpp*
- Scrivere la dichiarazione della classe nel file *compito.h* e la definizione delle funzioni nel file *compito.cpp*. Il file *main.cpp* contiene la funzione principale *main()* che serve a verificare le funzioni scritte nel file *compito.cpp*. Il file *main.cpp* può essere modificato. **Si tenga presente, comunque, che in sede di valutazione dell'elaborato verrà considerato esclusivamente il contenuto dei file *compito.h* e *compito.cpp***
- **Per una corretta stampa dell'elaborato**, non scrivere righe di codice di lunghezza eccessiva (mantenersi entro i margini imposti dall'ambiente *Dev-C++* cioè entro la linea verticale presente alla destra della pagina);
- **Per la Consegna:**
 - Selezionare la voce *Consegna* dal menù *Strumenti* (o *Tools*) all'interno dell'ambiente *Dev-C++* e premere il tasto INVIO fino a quando non viene chiusa la finestra che è stata attivata.
 - Aspettare al proprio posto di essere chiamati per verificare la stampa del proprio elaborato, firmarlo e **consegnarlo definitivamente**.

Condizione necessaria per la correzione dell'elaborato è che le prime tre funzioni siano state implementate correttamente.