

# PGMImage Class User Manual

Questa classe permette di leggere e scrivere immagini in formato PGM (Portable Gray Map) immagazzinando i dati in memoria per recuperarli e modificarli.

## **CPGMImage()**

*Descrizione:* Costruttore vuoto della classe, non alloca alcun spazio in memoria per l'area dati dell'immagine

*Parametri:* nessuno.

*Valore di ritorno:* nessuno.

## **CPGMImage(unsigned int width, unsigned int height)**

*Descrizione:* Costruttore della classe, alloca spazio in memoria riservato all'area dati dell'immagine, con dimensioni *width* per *height* pixel, inizializzando la memoria con il valore 0.

*Parametri:*

- *width* : dimensione sull'asse X dell'immagine (larghezza).
- *height* : dimensione sull'asse Y dell'immagine (altezza).

*Valore di ritorno:* nessuno.

## **~CPGMImage()**

*Descrizione:* Distruttore della classe, dealloca l'eventuale spazio in memoria riservato all'area dati dell'immagine.

*Parametri:* nessuno.

*Valore di ritorno:* nessuno.

## **Int getWidth()**

*Descrizione:* funzione che restituisce la larghezza dell'immagine.

*Parametri:* nessuno.

*Valore di ritorno:* la larghezza dell'immagine, nel caso in cui l'immagine non sia allocata (e quindi la dimensione non è valida) viene ritornato un valore d'errore pari a -1.

## **Int getHeight()**

*Descrizione:* funzione che restituisce l'altezza dell'immagine.

*Parametri:* nessuno.

*Valore di ritorno:* l'altezza dell'immagine, nel caso in cui l'immagine non sia allocata (e quindi la dimensione non è valida) viene ritornato un valore d'errore pari a -1.

### ***bool resize(unsigned int width, unsigned int height)***

*Descrizione:* ridimensiona l'immagine, deallocando l'eventuale memoria presente e alloca spazio in memoria riservato all'area dati dell'immagine, con dimensioni *width* per *height* pixel, inizializzando la memoria con il valore 0.

*Parametri:*

- *width* : nuova dimensione sull'asse X dell'immagine (larghezza).
- *height* : nuova dimensione sull'asse Y dell'immagine (altezza).

*Valore di ritorno:* flag che indica il successo dell'operazione, sarà *true* nel caso in cui l'immagine è stata correttamente ridimensionata, *false* nel caso in cui l'operazione non può essere eseguita (impossibile allocare la memoria).

### ***bool getPixel(unsigned int x, unsigned int y, unsigned char\* value)***

*Descrizione:* preleva il valore del pixel alla posizione (x,y).

*Parametri:*

- *x* : coordinata x del pixel da leggere.
- *y* : coordinata y del pixel da leggere.
- *value* : puntatore che indica la locazione di memoria in cui è presente la variabile in cui verrà scritto il valore del pixel letto.

*Valore di ritorno:* flag che indica il successo dell'operazione, sarà *true* nel caso in cui il valore letto alla locazione di memoria indicata da *value* è valido, *false* nel caso in cui l'operazione non può essere eseguita (in caso di memoria non allocata oppure di posizione richiesta al di fuori delle dimensioni dell'immagine).

### ***bool setPixel(unsigned int x, unsigned int y, unsigned char value)***

*Descrizione:* modifica il valore del pixel alla posizione (x,y).

*Parametri:*

- *x* : coordinata x del pixel da modificare.
- *y* : coordinata y del pixel da modificare.
- *value* : valore da scrivere nel pixel alla posizione (x,y).

*Valore di ritorno:* flag che indica il successo dell'operazione, sarà *true* nel caso in cui il valore scritto alla locazione di memoria indicata da *value* è valido, *false* nel caso in cui l'operazione non può essere eseguita (in caso di memoria non allocata oppure di posizione richiesta al di fuori delle dimensioni dell'immagine).

### ***bool loadPGM(char\* fileName)***

*Descrizione:* carica da un file PGM l'immagine riempiendone l'area dati (dopo aver opportunamente, se necessario, deallocato e reallocato l'area dati opportuna). La funzione riconosce automaticamente il formato dell'immagine se ASCII (indicato da un P2 in cima all'header dell'immagine) o RAW (indicato da un P2 in cima all'header dell'immagine).

*Parametri:*

- *fileName* : stringa contenente il nome del file da caricare.

*Valore di ritorno:* flag che indica il successo dell'operazione, sarà *true* nel caso in cui il file è stato letto e l'immagine è stata correttamente allocata, *false* nel caso in cui l'operazione non può essere eseguita (file illeggibile o inesistente, header o area dati non corretti).

### ***bool savePGM(char\* fileName, bool raw=false)***

*Descrizione:* salva in un file PGM l'immagine, scrivendo l'header corretto e l'area dati nel formato richiesto: il parametro *raw* è il flag che indica se scrivere l'area dati in formato ASCII (*false*) oppure RAW (*true*).

*Parametri:*

- *fileName* : stringa contenente il nome del file su cui scrivere l'immagine.
- *raw*: flag che indica il formato del file PGM in cui scrivere l'immagine, se è *true* l'immagine verrà scritta in formato RAW, altrimenti se *false* sarà scritta in formato ASCII.

*Valore di ritorno:* flag che indica il successo dell'operazione, sarà *true* nel caso in cui il file correttamente scritto, *false* nel caso in cui l'operazione non può essere eseguita (area dati non allocata o impossibile scrivere i file indicato).