

PROVA PRATICA DI INFORMATICA & LAB. PROGRAMMAZIONE

6 giugno 2022

Nome: _____ Cognome: _____ Matr: _____ Postazione _____

Scrivere un programma in linguaggio C (chiamare il progetto con la propria <matricola>) che abbia il comportamento descritto nel seguito. Il tempo a disposizione è di **120 minuti**. Al termine del tempo gli elaborati salvati su **U:** verranno raccolti automaticamente dal sistema di laboratorio. Eventuali documenti sono disponibili in **T:\Bertozzi**, si consiglia di usare **wordpad** per leggere i file di testo.

Sono forniti alcuni file con estensione “.pgm”. Questi file memorizzano una immagine in bianco e nero e hanno il seguente formato:

La prima riga contiene la stringa **P2** seguita da un “a capo”

La seconda riga contiene due numeri seguiti da un “a capo” che indicano, rispettivamente, larghezza **X** e altezza **Y** dell’immagine in punti.

La terza riga contiene **255** seguito da un “a capo”

A seguire ci sono **X*Y** numeri interi tra 0 e 255, separati da spaziature, che indicano quanto nero o bianco è ciascun punto.

Ad esempio:

P2

640 426

255

141 136 133 133 134 130 125 122 123 123 122 119 116 114 112 112
111 112 103 99 108 101 111 169 108 99 96 100 100 94 94 100 97...

Si sviluppi un programma, a partire dallo scheletro di codice fornito, che:

1. contenga una funzione `int *load_pgm(char *filename, int *sizex, int *sizey)` che carica i punti dell’immagine contenuta nel file “filename”, restituisce l’indirizzo di un’area di memoria allocata dinamicamente che contiene tali valori e deposita in *sizex e *sizey le dimensioni in punti dell’immagine
2. contenga una funzione `int *vedge(const int *dati, int cols, int rows, int th)` che prende in ingresso l’indirizzo di un’area di memoria che contiene i punti di una immagine caricata dalla `load_pgm()` e relative dimensioni cols/rows. Tale funzione alloca dinamicamente una seconda area delle stesse dimensioni di quella in input. Per ciascun punto della prima immagine calcola il valore assoluto (funzione `abs()` di `math.h`) della differenza tra il punto precedente e quello seguente e lo memorizza in questa seconda area se e solo se il risultato risulta $\geq th$. Altrimenti vi mette 0. Per semplicità, si considerino come punto precedente e successivo quelli che seguono e precedono il punto in esame nell’array
Tale funzione restituisce l’indirizzo dell’area allocata dinamicamente.
3. contenga una funzione `void save_pgm(const char *filename, const int *dati, int cols, int rows)` che salva in un file di nome “filename” l’immagine i cui punti sono memorizzati all’indirizzo “dati” di dimensione cols*rows con lo stesso formato precedentemente descritto