

Per chi non ricevesse le email sulla mailing list, ecco alcuni suggerimenti per l'esecuzione del test (29/12/2003).

Ciao,

ho iniziato a vedere alcuni risultati dei test. Vi scrivo alcuni consigli utili. La mail e' piuttosto lunga ma vorrei la leggeste, vi aiuta a capire se state risolvendo bene l'esercizio.

Test 1.

- Non dimenticate di mandarmi i grafici dei dati che ottenete. Potete usare matlab, excel o quello che preferite.

- Dividete il codice in funzioni !! Sarebbe opportuno avere una funzione di lettura dei dati, e una per ogni interpolazione (NN e lineare, 2x e 3x).

Se avete dei problemi a ritornare i valori vi spiego come fare:

a) ritornate il puntatore all'area dati dalla funzione di lettura. L'allocazione in questo caso avviene dentro alla funzione (ricordate di deallocare alla fine del main). Come parametro potete ritornare il numero dei dati:

Es prototipo:

```
double *leggiDati(const char *nomefile, int *n)
```

b) allocate nel main e fate riempire il vettore dalla funzione di lettura. In questo caso vi serve una lettura anticipata del file per determinare la quantita' dei dati (non infilate nel codice il valore '10', dovete leggerlo da file). In questo caso potete fare una semplice funzione che apre il file, legge il primo numero, chiude il file e restituisce il numero letto.

Es prototipi:

```
int leggiNumeroDati(const char *nomefile);
```

```
void leggiDati(const char *nomefile, double *dati, int n);
```

c) passate il puntatore al puntatore, questo caso e' simile al primo

```
void leggiDati(const char *nomefile, double **dati, int *n);
```

Le funzioni per l'interpolazione sono simili; in questo caso sapete a priori la dimensione dei vettori quindi e' tutto un po' piu' facile perche' potete allocare la memoria nel main (e ovviamente deallocarla alla fine).

Test 2.

Le funzioni membro zoombilinear() zoomNN() non devono scrivere su disco. Vorrei che tutte le operazioni venissero fatte in memoria; dovete creare un'area dati temporanea tale da contenere l'immagine di dimensioni doppie (potete anche creare un oggetto di tipo immagine se preferite), riempire i pixel in modo opportuno (secondo le formule dell'interpolazione relativa) e fare la sostituzione con l'area dati esistente.

Il main del test2 dovrebbe essere una cosa tipo:

```
main()  
{
```

```
immagine test1;
immagine test2;

test1.loadPPM("test.ppm");
test2.loadPPM("test.ppm");

test1.zoomNN();
test2.zoombilinear();

test1.savePPM("nn.ppm");
test2.savePPM("bilinear.ppm");
}
```

Va bene anche "cablare" il nome dei file come ho fatto io nell'esempio. Sconsiglio cose piu' complicate (tipo programmi che richiedono di inserire il nome del file, oppure se si vuole eseguire l'interpolazione lineare o quella bilineare), non sono necessarie.

Spero di essere stato chiaro, altrimenti... chiedete.

ciao,
Lorenzo

You're receiving this message because you are part of the SINA students' list.
Please contact postmaster@lira.dist.unige.it for further information.