

Laboratorio di Algoritmi e Strutture Dati

Esercitazioni del 13 Novembre 2012

Esercizio 4: Scritte LED

Si richiede di implementare un programma che:

1. Stampi il messaggio:

`Immetti numero:`

2. Legga un numero intero ≥ 0 (non si richiede di implementare controlli)
3. Stampi il numero nello stile degli orologi digitali a LED, come nell'esempio.

Esempio:

Supponiamo che l'utente immetta il numero 1536. Allora il programma deve stampare:

```
  =  =  =  
||    ||  
  =  =  =  
|  |  || |  
  =  =  =
```

Si suggerisce di predisporre un array `static char digit[]` inizializzato con valori 0 e 1 nel modo suggerito qui di seguito:

- Si noti che la rappresentazione LED di una cifra c accende un sottoinsieme $S(c)$ dell'insieme totale di 7 barrette orizzontali o verticali. In particolare la cifra 8 accende tutte e 7 le barrette.
- Dunque ogni cifra c può essere rappresentata come una sequenza di 7 valori booleani.
- Memorizzate nell'array `digit` tutte le 10 sequenze di 7 valori booleani corrispondenti alle 10 cifre.
- Al momento di stampare (una parte del)la cifra c , consultate le posizioni corrispondenti nell'array `digit` per determinare quali barrette accendere.

Nota:

Si consiglia di convertire il numero letto in una stringa. Per costruire questa stringa si può procedere in diversi modi. Ne segnaliamo due:

1. Leggere il numero carattere per carattere, utilizzando magari qualche funzione implementata in esercizi delle lezioni precedenti.
2. Leggere il numero con `scanf` e convertirlo con la funzione `sprintf` della libreria standard (prototipo in `stdio.h`). La funzione `sprintf` funziona come `printf` ma ha un argomento addizionale, il primo, che è l'indirizzo base di un array di `char`, dove memorizzare la stringa creata al posto di visualizzarla sullo schermo.

Esempio:

```
sprintf(s, "Ho %d gatti", 5);
```

memorizza in `s` la stringa `Ho 5 gatti`. L'argomento `char *s` deve essere (il nome di) un array di dimensione sufficiente a contenere la stringa risultato.

Sviluppo

Estendere il programma in modo da gestire anche i numeri interi negativi.

Esempio:

Supponiamo che l'utente immetta il numero -409 . Allora il programma deve stampare:

```
      =  =  
    |  |  |  |  
=  =  =  =  
    |  |  |  |  
      =  =
```

Facoltativo

Ogni combinazione di 7 bit può essere contenuta in un `char`, dunque si può sostituire all'array `char digit[]` che contiene $7 * 10$ caratteri, un array `char digit1[]` di soli 7 caratteri.

Implementate il programma precedente utilizzando `digit1` al posto di `digit`.