

Laboratorio di Algoritmi e Strutture Dati

Esercitazioni del 30 Ottobre 2012

Esercizio 1: automa cellulare monodimensionale

Si consideri un array `line` di dimensione `LINEDIM` pari a 60, i cui elementi detti *celle* possano assumere solo due valori: 0 o 1.

Se la cella `line[i]` vale 1 allora diciamo che è *piena*, altrimenti, diciamo che è *vuota*.

L'array `line`, detto *automa cellulare monodimensionale*, evolve nel tempo per mezzo di una *regola*, nel modo che andiamo a descrivere.

1. La *regola* è codificata da un numero intero $0 \leq c \leq 255$.
(Esempio: $c = 30$)
2. Si consideri la rappresentazione binaria β_c di c (vedi esercizio `dec2bin`, lezione del 16 Ottobre) e la si espanda premettendo tanti zeri iniziali quanti ne servono per fare in modo che β_c sia una stringa di 8 bit
(Esempio: $\beta_{30} = 00011110$)
3. A ogni terna di bit γ si associ a questo punto il valore 0 o il valore 1 in questo modo:
sia d il valore (in base 10) di γ (Esempio: se $\gamma = 110$ allora $d = 4 + 2 = 6$) allora a γ si deve associare il d -esimo bit di β_c
(Esempio: $\beta_{30} = 00011110$ determina le seguenti associazioni:

111 : 0 110 : 0 101 : 0 100 : 1 011 : 1 010 : 1 001 : 1 000 : 0

)

4. Si Assuma che lo stato dell'automa (vale a dire il valore di tutte le celle) al momento t sia registrato nell'array `line`. Allora lo stato dell'automa al momento $t + 1$ deve essere calcolato nel modo di seguito descritto:
 - Per ogni valore dell'indice $0 \leq i < \text{LINEDIM}$, il valore al tempo $t+1$ di `line[i]` è completamente determinato dal valore al tempo t della terna di bit
 $(\text{line}[i-1], \text{line}[i], \text{line}[i+1])$.
(nel caso $i == 0$, dalla terna $(\text{line}[\text{LINEDIM}-1], \text{line}[0], \text{line}[1])$,)
(nel caso $i == \text{LINEDIM} - 1$ dalla terna $(\text{line}[\text{LINEDIM}-2], \text{line}[\text{LINEDIM}-1], \text{line}[0])$).
 - Sia (b_1, b_2, b_3) la terna di bit che determina il valore di `line[i]` al tempo $t+1$. Allora il valore da assegnare a `line[i]` al tempo $t+1$ è il valore che la regola c associa alla terna (b_1, b_2, b_3)
(Esempio, se $c = 30$ e $(b_1, b_2, b_3) = (1, 1, 0)$ allora `line[i] = 0`)

5. L'evoluzione nel tempo dell'automa, a partire da un dato array `line` al tempo $t = 0$, si deve visualizzare su schermo stampando lo stato dell'automa al tempo $t+1$ sulla riga successiva a quella in cui si è stampato l'automa al tempo t .
(Esempio: visualizzando ogni cella piena con '*' e ogni cella vuota con uno spazio bianco, ecco l'evoluzione dal tempo $t = 0$ al tempo $t = 20$ di un automa cellulare monodimensionale per mezzo della regola 30 a partire dallo stato iniziale ($t = 0$):

11011100101011111100100101100100011010000100100011001100111

```

** *** * * ***** * * ** * ** * * * ** ** ***
   * *** * * ***** * **** ** ** ***** ** *** ***
   ***** * ** ** * * * **** *** * * * *
**   * ** * * ** * ** ** * ** * * * ***** *****
* * ** * * ***** ** ** * **** ** ***** **
** ***** ***** * **** *** **** * **** * ****
* * *** ** ** * * * **** ***** ** ** ** * *
***** ** * ** * * **** **** ** **** * ** * ** *****
*   * ***** *** * * * * * ** * ** * * * *
**** ** *   * * ** **** ** * * **** * **** *****
* *** ** ***** ** * * * * * **** * **** * * *
**** *** * ** * ***** ***** * ** * * ****
* **** ***** * ***** *   * * * * ***** * **
***** * ** * * * * **** ** ***** * * **** * *
*   *** * * ** ** **** ** * ** * * ** ** * ** *
* ** * ***** * **** * **** * **** ** * ** * *
***** * ** * ***** * * ** * * **** * * * ****
**   * * ** ** ***** * ** ** ** ** ** ** **** ** ***** *
   * ** ***** * * ** ***** * * * **** * * **
***** *   *** * * * ** ***** ***** * * ***** ** *
*   ***** ** * * ***** * ** ***** * **** ***

```

Si richiede di realizzare un programma che:

1. Richieda in input il numero intero n di iterazioni.
2. Richieda in input il codice numerico c della regola con cui fare evolvere l'automa.
3. Stampi su una nuova riga i valori binari associati alle terne di bit determinati dalla regola c , nel formato esemplificato qui di seguito:
Esempio: se $c = 110$ allora si deve stampare la riga:

```
[111:0] [110:1] [101:1] [100:0] [011:1] [010:1] [001:1] [000:0]
```

4. Generi un array di LINEDIM bit (lo si memorizzi in un array `char line[LINEDIM]`) in modo casuale, utilizzando le espressioni:

```

srand(time(NULL)) /* inizializza il generatore pseudocasuale */
rand() % 2 /* genera un bit pseudocasuale */

```

(per utilizzare le funzioni di libreria standard `srand` e `rand` bisogna includere `stdlib.h`, per utilizzare `time` bisogna includere `time.h`).

L'array `line` a questo punto contiene lo stato iniziale dell'automa (tempo $t = 0$).

5. Stampi su una nuova riga lo stato iniziale dell'automa usando '*' per le celle piene e lo spazio per le celle vuote.
6. Per ogni $1 \leq t \leq n$ (dove n è il numero di iterazioni immesso dall'utente), si calcoli lo stato dell'automa al tempo t e lo si stampi su una nuova riga usando '*' per le celle piene e lo spazio per le celle vuote.