



MR57 mangiasoldi

Giovanni A. Cignoni – Dipartimento di Informatica, Università di Pisa – marzo 2012

Sorgente commentato

È un generatore di numeri pseudocasuali congruenziale: $x_{i+1} = (ax_i + c) \bmod m$. Come è noto, i generatori di questo tipo hanno periodo m e generazione di tutti in valori in $[0 .. m-1]$ se m e c sono primi fra loro, se $a-1$ è divisibile da tutti i fattori primi di m , e se $a-1$ è multiplo di 4 quando m è multiplo di 4. Di conseguenza, una buona scelta di costanti è:

- modulo (m) 262144, 2^{18} , sfrutta l'overflow naturale, m è multiplo di 4;
- moltiplicatore (a) 129, 2^8+1 , permette di usare shift e somme, $a-1$ è multiplo di 4;
- incremento (c) 100003, primo.

Il programma parte a 100 (eseguire uno Z 100). Il seme (s) è per default fissato a 12345, ma può essere cambiato in 119 (eseguire un QM119). Il programma è un ciclo infinito, con interruzioni in:

- 112, CAC = 100, memorizzazione del numero generato;
- 115, CAC = 001, memorizzazione del numero mascherato.

La visualizzazione sull'indicatore della memoria (IM) del numero mascherato permette un uso come *slot machine*: come configurazioni vincenti valgono coppie e tris dei 3 bit centrali dei 3 gruppi di 5 bit di IM, il triplo 7 è la vincita massima (la gestione dei gettoni non è implementata).

```
# 100 strt 000 n+A seed    # load seed in A
# 101      000 AM  x      # copy seed in x to initialize sequence
# 102 loop 000 n+A x      # load x in A
# 103      000 Vs        # multiply 2
# 104      000 Vs        # multiply 4
# 105      000 Vs        # multiply 8
# 106      000 Vs        # multiply 16
# 107      000 Vs        # multiply 32
# 108      000 Vs        # multiply 64
# 109      000 Vs        # multiply 128
# 110      000 A+A x      # multiply 129
# 111      000 A+A c      # add increment
# 112 brk1 100 AM  x      # store x+1
# 113      000 n+B x      # load x in B
# 114      000 LpB mask   # light off some bits
# 115 brk2 001 BM  v      # store to visualize
# 116      000 Z  loop    # go to generate next number
# 117 x      000000       # generated pseudo random value
# 118 c      100003       # increment
# 119 seed   012345       # seed
# 120 mask   014798       # 000 01110 01110 01110
# 121 v      000000       # visualized value
```

Sorgente caricabile come immagine di memoria

100

22

000 n+A 119

000 AM 117

000 n+A 117

000 Vs 000

000 Vs 000

000 Vs 000

000 Vs 000

000 Vs 000

000 Vs 000

000 Vs 000

000 A+A 117

000 A+A 118

100 AM 117

000 n+B 117

000 LpB 120

001 BM 121

000 Z 102

000000

100003

012345

014798

000000