

5 5 5 1 6...

Numeri casuali

Marco Alberti



Dipartimento
di Matematica
e Informatica



Università
degli Studi
di Ferrara

Programmazione e Laboratorio, A.A. 2021-2022

Ultima modifica: 17 novembre 2021

Attenzione! Questo materiale didattico è per uso personale dello studente ed è coperto da copyright.
Ne sono vietati la riproduzione e il riutilizzo anche parziale, ai sensi e per gli effetti della legge sul diritto d'autore.

Generazione di numeri casuali

$\text{rand}() \rightarrow 18511$
 $\text{rand}() \rightarrow 525$

La funzione `int rand(void)`, dichiarata in `<stdlib.h>`, genera un intero casuale compreso fra 0 e `RAND_MAX` (una costante intera, almeno 32767).

Generazione di `int` casuali:

- L'espressione `rand() % 10` vale un numero casuale compreso fra 0 e 9.
- Più in generale, un numero intero casuale compreso fra `min` e `max` si ottiene da
 $\rightarrow \text{min} + \text{rand}() \% (\text{max} - \text{min} + 1)$

$1 + \text{rand}() \% 6 \rightarrow$ compreso fra 1 e 6

$15 + \text{rand}() \% (99 - 15 + 1)$
85

Generazione di `double` casuali:

- L'espressione `(double) rand() / RAND_MAX` vale un `double` casuale compreso fra 0.0 e 1.0.
- In generale, un `double` casuale compreso fra `min` e `max` si ottiene come
 $\rightarrow \text{min} + (\text{double}) \text{rand}() / \text{RAND_MAX} * (\text{max} - \text{min})$

365 405

$\frac{525}{1000} = 0.525$
 $36.5 + 4.0 \cdot \frac{(\text{double}) \text{rand}()}{\text{RAND_MAX}}$

Scrivere un programma che simuli 10 lanci di un dado a 6 facce (cioè stampi 10 numeri casuali compresi fra 1 e 6).

097_numeri_casuali/dado.c

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main(void) {
5      int i;
6      for (i = 0; i < 10; i++)
7          printf("%d\n", 1 + rand() % 6);
8      return 0;
9  }
```

Handwritten annotations:

- min = 1* (next to line 1)
- max = 6* (next to line 2)
- min* (with an arrow pointing to the `1` in line 7)
- max - min + 1* (with an arrow pointing to the `6` in line 7)

Scrivere un programma che simuli 10 rilevazioni di temperature comprese fra 15.0°C e 25.0°C.

097_numeri_casuali/temperature.c

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main(void) {
5      int i;
6      for (i = 0; i < 10; i++)
7          printf("%f\n", 15.0 + 10.0 * rand() / RAND_MAX);
8      return 0;
9  }
```

Handwritten annotations:

- $\text{min} = 15.0$ (next to line 1)
- $\text{max} = 25.0$ (next to line 2)
- (double)* (next to line 6)
- min (under 15.0 with an arrow pointing to it)
- (max-min) (under 10.0 with an arrow pointing to it)

Funzioni per generazione di numeri casuali

097_numeri_casuali/funzioni-casuali.c

```
1  int rnd_int(int 1min, int 6max) {  
2      return min + rand() % (max - min + 1);  
3  }  
4  
5  double rnd_double(double 15.0min, double 25.0max) {  
6      return min + (double)rand() / RAND_MAX * (max - min);  
7  }
```

```
main() {  
    . rnd_int(1,6) ..  
}
```

PSEUDO CASUAL

$$r: \mathbb{N} \rightarrow \mathbb{N}$$

$$r_1 \quad \text{fixo} \quad \text{SEED} (1) (100)$$

$$r_2 = f(r_1)$$

$$r_3 = f(r_2)$$

$$r_n = f(r_{n-1})$$

- La sequenza di numeri casuali restituiti da `rand()` in un programma è in realtà una sequenza di numeri pseudo-casuali calcolati ognuno dal precedente, a partire da un numero detto **seed** (seme).
- Se non impostato, il seed è 1. Quindi esecuzioni diverse di uno stesso programma generano la stessa sequenza di numeri casuali!
- Per impostare il seed, si può usare la funzione `void srand(unsigned int 100 seed);`. Ad esempio, l'istruzione `srand(100);` imposta il seed a 100.
- In questo modo però esecuzioni diverse dello stesso programma genereranno ancora sequenze uguali di numeri casuali (anche se diverse da quelle di seed 1).
- Per avere sequenze diverse, si può impostare il seed a un valore che varia necessariamente fra un'esecuzione e l'altra, ad esempio quello restituito dall'espressione `time(NULL)` (numero di secondi trascorsi dal 1 gennaio 1970).

`time(NULL)` → 1'500'000'000
→ 1'500'000'005

Simulazione di 10 lanci di un dado, con risultato che varia da esecuzione a esecuzione:

097_numeri_casuali/dado-seed.c

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(void) {
5     int i;
6     srand(time(NULL));
7     for (i = 0; i < 10; i++)
8         printf("%d\n", 1 + rand() % 6);
9     return 0;
10 }
```

(IMPOSTAZIONE SEED)

Seed = 100
rand() → 5000
~~Seed = 100~~
rand() → 5000