

# Números aleatorios

Taller de Lenguajes 1

# Generación de números aleatorios

- La función **rand()** genera un número entero entre 0 y **RAND\_MAX** con distribución uniforme.
- En ANSI C, el valor de **RAND\_MAX** debe ser por lo menos 32767 (16 bits). Está definida en `<stdlib.h>`
- En las implementaciones actuales el valor **RAND\_MAX** suele ser 4294967295 (32 bits).
- Para acotar los valores posibles devueltos por la función **rand()** debe usarse la función módulo **%**.
- Ejemplos
  - **NumeroDelDado = 1 + rand() % 6**
  - **NroDeMes = 1 + rand() % 12**

# Ejemplo

- Escriba un programa C que simule tirar una moneda 7000 veces. Al finalizar debe informar cuantas veces salió cara y cuantas veces salió ceca.
- Ejecute varias veces el mismo programa y observe los resultados obtenidos.

## 02\_LanzaMoneda.c

```
/* Lanza una moneda 7000 veces */  
#include <stdio.h>  
int main()  
{   int i, cara=0;  
  
    for (i=0; i<7000; i++)  
        if (rand() % 2) /* no es par */  
            /* los números impares se cuentan  
               como cara */  
            cara++;  
  
    printf("Salieron %d caras y ", cara);  
    printf("%d cecas\n", 7000-cara);  
    return 0;  
}
```

Cada vez que lo ejecuto imprime  
«Salieron 3521 caras y 3479  
cecas»

# Función **srand**

- La función **rand** genera números pseudoaleatorios.
- Para que la generación de números sea aleatoria debe usarse la función **srand**.
- La función **srand** tiene un argumento de tipo unsigned, llamado semilla y permite que en cada ejecución del programa la función rand genere una secuencia de números aleatorios diferentes.
- El tipo **unsigned** es equivalente a **unsigned int** y representa un entero sin signo.

```
/* Lanza una moneda 7000 veces */
#include <stdio.h>
int main()
{   int i, cara=0;
    unsigned semilla;

    printf("Ingrese el valor de la semilla : ");
    scanf("%u", &semilla);

    srand(semilla);
    for (i=0; i<7000; i++)
        if (rand() % 2) /* no es par */
            /* los números impares se cuentan
               como cara */
            cara++;

    printf("Salieron %d caras y ", cara);
    printf("%d cecas\n", 7000-cara);
    return 0;
}
```

## 02\_LanzaMoneda1.c

```
/* Lanza una moneda 7000 veces */
```

```
#include <stdio.h>
```

```
int main()
```

```
{ int i, cara=0;
```

```
    unsigned semilla;
```

## 02\_LanzaMoneda1.c

- Un **unsigned** o **unsigned int** de dos bytes puede tener sólo valores positivos dentro del rango 0 hasta 65535.
- Un unsigned de 4 bytes puede tener sólo valores positivos entre 0 y 4294967295

```
printf("Salieron %d caras y ", cara);
```

```
printf("%d cecas\n", 7000-cara);
```

```
return 0;
```

```
}
```

```
/* Lanza una moneda 7000 veces */
```

```
#include <stdio.h>
```

```
int main()
```

```
{  int i, cara=0;
```

```
    unsigned semilla;
```

```
    printf("Ingrese el valor de la semilla : ");
```

```
    scanf("%u", &semilla);
```

## 02\_LanzaMoneda1.c

- Dentro de **scanf** utilice **"%u"** para leer un **unsigned**.

```
}
```



```
/* Lanza una moneda 7000 veces */
#include <stdio.h>
int main()
{   int i, cara=0;
    unsigned semilla;

    printf("Ingrese el valor de la semilla : ");
    scanf("%u", &semilla);

    srand(semilla);
    for (i=0; i<7000; i++)
        if (rand() % 2)    /* no es par */
```

## 02\_LanzaMoneda1.c

- La función **srand** recibe la *semilla* indicada.
- Luego, la secuencia generada por la función rand cambiará cada vez que cambie la semilla (para la misma semilla la secuencia es la misma)

```
}
```

# Función time

- Para no tener que indicar el valor de la semilla en cada ejecución puede utilizarse la función **time** de la siguiente forma

**srand(time(NULL))**

- La función time devuelve la hora actual del día en segundos. Ese valor es convertido a un unsigned y utilizado como semilla.
- El argumento NULL hace que la función no devuelva ningún valor a través de la lista de parámetros.
- La función time está definida en <time.h>