

## 8. Librerías

Programación en C.

# Librería math.h

---

- Contiene funciones útiles cuando queremos realizar operaciones matemáticas. Hay que incluirla:

```
#include <math.h>
```

- Para enlazarla:

```
$gcc -o nombreprograma nombreprograma.c -lm
```

Entre las funciones incluidas en la librería `math.h` están:

- Logaritmo neperiano de `x`: `double log(double x);`
- Logaritmo en base 10 de `x`: `double log10(double x);`
- Potencia `x` elevado a `y`: `double pow(double x, double y);`
- Raíz cuadrada de `x`: `double sqrt(double x);`
- Redondear hacia arriba el número `x`: `double ceil(double x);`
- Redondear hacia abajo el número `x`: `double floor(double x);`
- Valor absoluto: `double fabs(double x);`
- Seno de `x`: `double sin(double x);`
- Coseno de `x`: `double cos(double x);`
- ▶ – Tangente de `x`: `double tan(double x);`

# Librería `stdlib.h`

---

- Contiene funciones útiles cuando queremos obtener un número aleatorio.
- Los números aleatorios se usan en programación para simular por ejemplo el resultado de tirar un dado, distribuir los premios ocultos en un juego (arkanoid), inicializar una jugada (posición de barcos en hundir la flota), etc.
- Mediante la función `rand()` obtenemos un número aleatorio entre 0 y el valor especificado por la constante `RAND_MAX`. El valor de la constante `RAND_MAX` depende del compilador, pero como mínimo vale `0x7FFF`.
- Para obtener un número aleatorio entre 0 y N, tendríamos que hacer lo siguiente: `num_aleatorio=rand() % (N+1);`



# Librería `stdlib.h`

---

- La función `rand()` calcula el número aleatorio usando un algoritmo que parte de un número (semilla) y a partir de él calcula los números aleatorios.
- Para que los números aleatorios sean distintos, hay que inicializar la semilla:  
`srand()` asigna una semilla (la que se le pase como parámetro).
- El número pasado a `srand()` no puede ser un valor fijo: se le pasa un valor que sea distinto en cada ejecución del programa, como por ejemplo la hora/fecha del sistema (cambia en cada instante).



# Librería time.h

---

- Contiene funciones relacionadas con la hora del sistema.
- La función `time()` proporciona la hora del sistema



# Para generar un número aleatorio...

---

- Se incluyen las librerías `stdlib.h` y `time.h`
- Se incluye una sola vez en el programa la siguiente instrucción:  
`srand(time(NULL));`  
Así se inicializa la semilla para generar números aleatorios
- Para generar un número aleatorio:  
`num_aleatorio=rand() % (N+1);`



# Programa 1

---

Realice un programa *adivina.c* en lenguaje C que implemente el juego “Adivina el número”, que consiste en introducir cifras por parte del usuario a nuestro programa hasta que acertemos el número.

- La función principal generará un número aleatorio entre 0 y 100 y se lo pasará a una función *adivina*. Ésta le devolverá el número de intentos y la función principal los imprimirá por pantalla.
- La función *adivina* leerá los enteros que le vaya diciendo el usuario e indicará si el entero leído es mayor o menor al número a adivinar (que se le pasa como argumento). Si acierta, se imprimirá la cadena “Acertaste!!” y devolverá el número de intentos que ha necesitado para adivinar el número.



# Programa 2

---

Realice un programa *intervalo.c* en lenguaje C que implemente el siguiente programa:

- La función principal generará dos números aleatorios entre 5 y 25 y se los pasará a una función *intervalo*.
- La función *intervalo* imprimirá todos los números impares que hay desde el aleatorio mayor hasta el aleatorio menor. Es decir, si los argumentos son 7 y 14, imprimirá 7, 9, 11 y 13. Si son 14 y 10 imprimirá 11 y 13.

