

# Curso de games FK em C – Aula 1

- Biblioteca time.h
- Função Rand, Srand e Seed
- Números aleatórios
- Rolando dado
- Cara e coroa

# Biblioteca time.h

- Fornece funções, macros e definições de tipos para manipular datas e horários
- A função time é usado como seed na chamada do comando srand para gerar uma semente imprevisível e variável.

# rand()

- Cada chamada da função produz um número aleatório no intervalo fechado 0..RAND\_MAX.
- A constante RAND\_MAX está definida na biblioteca stdlib.h. Faixa varia de 0 a (possivelmente) 32767 (dependendo de implementação pode ter maior amplitude).
- Esta função utiliza uma “semente” para gerar sua sequência aleatória, portanto, se chamada sem a criação de seed - uma semente, diferente a cada execução, seu resultado será sempre o mesmo, isto é, sempre iniciará com o mesmo número “aleatório” e apresentará a mesma sequência numérica a cada chamada, portanto, não esqueça, sempre chame a função rand após a função srand.

# Sintaxe - rand()

- Para gerar valores inteiros aleatórios em C no intervalo [a, b):

$a + \text{rand()} \% (b - a)$

a é o valor que iniciará a contagem aleatória

b o número final da contagem aleatória

c será o valor de  $c = a - 1$

# Exemplo 1

1 - Faça um programa para gerar 6 números aleatórios da mega sena de 1 a 59 usando a função rand().

```
#include <stdio.h>
#include <stdlib.h> // necessário p/ as funções rand() e srand()

//a + rand() % (b - c)
//a = 1 - número inicial
//b = 59 - número final
//c = 1 - 1 = 0.

int main()
{
    int i;

    for (i = 1; i <= 6; i++)
    {
        printf("%d ", 1 + rand() % (59 - 0));
    }

    return 0;
}
```

```
42 1 22 10 54 31
```

```
-----
```

```
Process exited after 0.07867 seconds with return value 0
```

```
Pressione qualquer tecla para continuar. . .
```

# srand()

- A função srand seria a abreviatura de “seed random” ou semente aleatória
- Configura o ponto de início para produzir uma série de números inteiros pseudoaleatórios
- Se srand() não for usado antes de rand() , o programa sempre gerará a mesma sequência de números a cada execução.

- Para inicializar o gerador de números aleatórios, a escolha entre `srand(time(0))` e `srand(time(NULL))` geralmente não faz diferença. Ambos são métodos válidos para gerar uma semente aleatória baseada no tempo atual. A diferença entre eles é apenas a forma como o valor do tempo é obtido, mas em essência, ambos fazem a mesma coisa.

1 - A função `time(0)` retorna o número de segundos desde a "época" (geralmente 1º de janeiro de 1970, também chamada de Unix epoch).

- O argumento 0 passado para `time()` significa que estamos pedindo o tempo atual. O valor retornado é um número inteiro representando o número de segundos desde a época

- 2 - A função `time(NULL)` é equivalente a `time(0)` — ela retorna o número de segundos desde a época.
- - `NULL` é simplesmente o valor que indica que o parâmetro de `time()` é ignorado. Em C, o uso de `NULL` é mais comum como uma maneira explícita de indicar um ponteiro nulo, mas, neste caso, o efeito é o mesmo que o uso de 0.
- - Ambas as formas são válidas e funcionam de maneira idêntica. O uso de `time(NULL)` é mais comum porque `NULL` é mais expressivo (indicando um ponteiro nulo) e geralmente é mais fácil de entender em códigos mais modernos.

## Exemplo 2

- Faça um programa para gerar 6 números aleatórios da mega sena de 1 a 59 usando a função `rand()` e `srand()`.

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

int main()
{
    int i;
    srand (time(NULL));

    for (i = 1; i <= 6; i++)
    {
        printf("%d ", 1 + rand() % (59 - 0));
    }

    return 0;
}
```

```
25 7 40 4 2 20
```

```
-----  
Process exited after 0.03599 seconds with return value 0  
Pressione qualquer tecla para continuar. . .
```

```
22 27 50 3 2 24
```

```
-----  
Process exited after 0.03581 seconds with return value 0  
Pressione qualquer tecla para continuar. . . |
```

# Números aleatórios

A geração de números aleatórios é importante em programação para diversas aplicações, como:

- Desenvolvimento de jogos
- Geração de senhas e textos de campos captcha
- Mineração de dados
- Análise estatística
- Simulações
- Geração de chaves criptográficas
- Geração de dados de teste de programas

# Exemplo 3

3 – Crie 2 matrizes  $3 \times 3$  e preencha a tabela usando número aleatórios de 10 a 19 e 1 a 9(incluindo esses números). Depois imprima as matrizes.

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
int main()
{
    int i,j;
    int matriz [3][3];
    srand (time(NULL));
    for (i = 0; i < 3; i++)
    {
        for (j = 0; j < 3; j++)
        {
            matriz[i][j] = 10 + rand() % (19 - 9);
        }
    }

    for (i = 0; i < 3; i++)
    {
        for (j = 0; j < 3; j++)
        {
            printf("%d ", matriz[i][j]);
        }
        printf("\n\n");
    }

    printf("\n");

    for (i = 0; i < 3; i++)
    {
        for (j = 0; j < 3; j++)
        {
            matriz[i][j] = 1 + rand() % (9 - 0);
        }
    }

    for (i = 0; i < 3; i++)
    {
        for (j = 0; j < 3; j++)
        {
            printf("%d ", matriz[i][j]);
        }
        printf("\n\n");
    }
    return 0;
}
```

13 10 15

10 15 14

17 16 12

7 3 5

8 2 3

4 9 9

-----  
Process exited after 0.02788 seconds with return value 0

Pressione qualquer tecla para continuar. . . |

# Exemplo 4 – Rolando dados

4 – Nos jogos de RPG(Role Playing game) precisamos rolar dados de diferentes tipos para realizar ações. Crie um programa onde o usuário poderá escolher em um menu qual tipo dado (4, 6, 8, 10, 12 e 20 faces) irá lançar e imprima o resultado. Faça que o programa volte ao menu depois de imprimir a jogada e de a opção de sair do jogo.





# Exemplo 5 – Cara e coroa

Em um jogo de moedas, cara e coroa, jogue 3 vezes a moeda. Verifique em quantos lançamentos terá a sequência de todas as moedas com o lado “cara”. Imprima o resultado da tela e o recorde (lançamento com menos rodadas).



# Fim da aula 1 de games

- Autor: [Fernando Eduardo](#).
- Dúvidas encaminhe e-mail para:  
[dunano@outlook.com](mailto:dunano@outlook.com)