



Números Aleatórios

Prof. Rosana

Funções em C para a geração de números randômicos

- Fazem parte da biblioteca `<stdlib.h>`
- São elas:
 - `rand()` – gera números aleatórios entre 0 e `RAND_MAX` (32767)
 - `srand()` – estabelece a semente de geração de números aleatórios
 - A semente controla a geração de números pela função `rand()`.
 - Se esta semente não for modificada entre uma execução do programa e outra, o conjunto de números aleatórios gerado será sempre o mesmo.

Geração de números randômicos

A função `srand()` recebe como parâmetro um número, que pode ser um valor constante, mas...

a forma **mais usual** de se usar o `srand()` é *variar a semente* a partir da hora de execução do programa:

Isto se consegue com a seguinte instrução:

```
srand((unsigned)time(NULL)); //ou  
srand(time(NULL));
```

OBS: Esta instrução vai aparecer uma única vez, no início do programa.

(*) Só que para usar a função `time()` é necessário incluir no início do programa `#include <time.h>`

Geração de números randômicos

Já, a forma mais usual da função `rand()` é converter o número gerado por ela em uma faixa de valores:

Exemplos:

`rand()%50` – gera números entre 0 e 49

`rand()%100` – gera números entre 0 e 99

Então, como conseguir números nos intervalos:

1 a 10 ??

10 a 50??

Geração de números randômicos

Respostas:

1 a 10 ??

`rand()%10 + 1`

10 a 50??

`rand()%41 + 10`

Vetores de tamanho variável

É possível criar um vetor de tamanho variável?

Considere:

```
int main() {  
    int i, tam;  
    printf("Informe o tamanho do vetor: ");  
    scanf("%d", &tam);  
    int vet[tam]; //???  
    for (i=0; i<tam; i++) {  
        vet[i]=i+1;  
        printf("%d\t", vet[i]);  
    }  
    printf("\n\n");  
    system("pause");  
}
```

Vetores de tamanho aleatório

Também é possível determinar randomicamente o tamanho o vetor

Considere:

```
int main() {
    srand(time(NULL));
    int i, tam;
    tam=rand()%20+1 // tamanho entre 1 e 20
    int vet[tam];
    for(i=0;i<tam;i++) {
        vet[i]=i+1;
        printf("%d\t",vet[i]);
    }
    printf("\n\n");
    system("pause");
}
```

Preenchimento do vetor com números aleatórios

O vetor será preenchido com números entre 1 e 50

Considere:

```
int main() {  
    srand(time(NULL));  
    int i, tam;  
    tam=rand()%20+1; // tamanho entre 1 e 20  
    int vet[tam];  
    for(i=0;i<tam;i++) {  
        vet[i]= rand()% 50+1;  
        printf("%d\t",vet[i]);  
    }  
    printf("\n\n");  
    system("pause");  
}
```


Preenchimento do vetor com números aleatórios

O preenchimento com números aleatórios **pode causar a repetição de valores**, especialmente se o numero de elementos for grande e a faixa de valores gerados pequena.

```
int main(){
    srand(time(NULL));
    int i, tam;
    tam=rand()%10+1 // tamanho entre 1 e 10
    int vet[tam];
    for(i=0;i<tam;i++) {
        vet[i]= rand()% 10+1;
        printf("%d\t",vet[i]);
    }
    printf("\n\n");
    system("pause");
}
```

Então, as vezes, é preciso evitar que o vetor seja preenchido com números repetidos!!

Para tanto deve-se verificar, a cada sorteio, se o valor gerado já se encontra no vetor.

Se isto ocorrer, o mesmo deverá ser descartado, senão armazenar o valor gerado !

Exercícios com vetores

1. Preencher um vetor de tamanho variável (mínimo 1 e máximo 30) com valores também variáveis na faixa de 1 a 30.
2. Escreva uma função do tipo void que escreva a posição de um elemento no vetor sempre que o mesmo for encontrado (podem ocorrer repetições). Caso não haja nenhuma ocorrência do elemento procurado, a função deve informar.
`void buscaRepetidos(int v[], int valor, int tam)`
3. Preencher um vetor de tamanho 10 com valores também variáveis na faixa de 1 a 10, mas sem repetição. (tam=10)
`void semRepetidos(int v[], int tam)`
4. Escreva uma função que efetue a busca de um determinado elemento em um vetor preenchido sem repetição. Se o elemento for encontrado a função deve retornar o índice do elemento, caso contrario -1 .
`int buscaElemento(int v[], int valor, int tam)`