

## Estrutura de repetição

- 1 Estruturas de Repetição
- 2 Estruturas de Repetição: WHILE
- 3 Condições (if) + Repetições (while)
  - Repetição dentro da condição
  - Condição dentro de repetição
  - Repetição dentro de repetição
- 4 Estrutura de Repetição: FOR

## O que é?

- Repetir uma mesma sequência de instruções até que uma condição seja satisfeita;
- Chamamos de LAÇO (loop);
- Ideia do “faça isso ATÉ QUE” ou “ENQUANTO isso acontecer faça”.

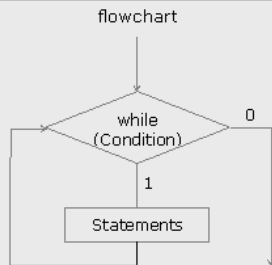
- 1 Estruturas de Repetição
- 2 Estruturas de Repetição: WHILE
- 3 Condições (if) + Repetições (while)
  - Repetição dentro da condição
  - Condição dentro de repetição
  - Repetição dentro de repetição
- 4 Estrutura de Repetição: FOR

# Estrutura de Repetição em C: WHILE

## Estrutura Básica

Syntax:

```
while(Condition)
{
  -----
  -----
  -----
}
```



# Estrutura de Repetição: WHILE

## Estrutura Básica - Atualização por atribuição

```
1 //variavel de controle
2 //declaracao
3 int i;
4
5 //inicializacao
6 i=0;
7
8 //repeticao
9 while ( i < 10 )
10 {
11     ...
12     //atualizacao por atribuicao i = i+1
13     i++;
14 }
15
```

- 1 Inicialização das condições
- 2 Repetição: enquanto condições forem verdadeiras faça ...
- 3 Atualização das condições por atribuição
  - ▶ Contador

# Estrutura de Repetição: WHILE

## Estrutura Básica - Atualização por atribuição

Exemplo: programa que imprime de 0 a 4

```
1 #include <stdio.h>
2
3 int main( )
4 {
5     int cont = 0;
6     printf("contador = %d\n", cont);
7
8     cont = cont+1;
9     printf("contador = %d\n", cont);
10
11    cont = cont+1;
12    printf("contador = %d\n", cont);
13
14    cont = cont+1;
15    printf("contador = %d\n", cont);
16
17    cont = cont+1;
18    printf("contador = %d\n", cont);
19
20    return 0;
21 }
```

# Estrutura de Repetição: WHILE

## Estrutura Básica - Atualização por atribuição

Exemplo: programa que imprime de 0 a 4

```
1 #include <stdio.h>
2
3 int main( )
4 {
5
6     //declarar e inicializar
7     int cont = 0;
8
9     //repetir - verificar
10    while( cont <= 4 )
11    {
12        //tarefa
13        printf( "contador = %d\n", cont );
14
15        //atualizar
16        cont = cont + 1;
17    }
18
19    return 0;
20 }
21
```

**E se fosse de 0 a 1000?**



## Estrutura de Repetição: WHILE

O que vai acontecer?!

```
1  while( 1 )
2  {
3      printf( "Ola mundo!\n");
4  }
5
6  int i = 0;
7  while(i < 5)
8  {
9      printf("%d\n", i+1);
10 }
11
```

Lembrando dos passos básicos:

- 1 Inicialização das condições
- 2 Repetição (verificação das condições)
- 3 Atualização das condições

## Estrutura de Repetição: WHILE

### Vamos praticar

Faça um programa que imprima a tabuada do 9 usando laço.

Saída:

9x0=0

9x1=9

...

9x10=90

```
1
2  inicializacao das condicoes
3
4  while ( condicoes )
5  {
6      ...
7      atualizacao das condicoes
8  }
```

# Estrutura de Repetição: WHILE

## Vamos praticar

```
1 #include <stdio.h>
2
3 int main()
4 {
5     //inicializar condicao - variavel de controle
6     int i=0;
7
8     //repetir - enquanto i for menor ou igual a 10
9     while(i <= 10)
10    {
11        ...
12
13        //atualizar
14        i++;
15    }
16
17    return 0;
18 }
19
```

# Estrutura de Repetição: WHILE

## Vamos praticar

```
1 #include <stdio.h>
2
3 int main()
4 {
5     //declarar e inicializar
6     int i=0;
7
8     //repetir - verificar
9     while(i <= 10)
10    {
11        //tarefa
12        printf("9x%d = %d\n", i, 9*i);
13
14        //atualizar
15        i++;
16    }
17
18    return 0;
19 }
20
```

## Vamos praticar

URI Online Judge | 1095

### Sequencia IJ 1

Adaptado por Neilor Tonin, URI  Brasil

**Timelimit: 1**

Você deve fazer um programa que apresente a sequencia conforme o exemplo abaixo.

#### Entrada

Não há nenhuma entrada neste problema.

#### Saída

Imprima a sequencia conforme exemplo abaixo

#### Exemplo de Entrada

#### Exemplo de Saída

```
I=1 J=60  
I=4 J=55  
I=7 J=50  
...  
I=? J=0
```

# Estrutura de Repetição: WHILE

## Vamos praticar

```
1 #include <stdio.h>
2
3 int main( ){
4
5     int i, j;
6
7     i = 1;
8     j = 60;
9     while (j >= 0)
10    {
11        printf("I=%d J=%d\n", i, j);
12        i=i+3;
13        j=j-5;
14    }
15    return 0;
16 }
17
```

# Estrutura de Repetição: WHILE

## Estrutura Básica - Atualização pela entrada padrão

```
1 //declaracao
2 int i;
3
4 //inicializacao
5 scanf("%d", &i);
6
7 //repeticao
8 while ( i != 0 )
9 {
10     ...
11     //atualizacao
12     scanf("%d", &i);
13 }
14
```

- 1 Inicialização das condições
- 2 Repetição: enquanto condições forem verdadeiras faça ...
- 3 Atualização das condições pela entrada padrão

# Estrutura de Repetição: WHILE

## Vamos praticar

Faça um programa que leia números da entrada padrão e imprima sua divisão por 2 enquanto o número lido for positivo.

```
1
2  inicializacao das condicoes
3
4  while ( condicoes )
5  {
6      ...
7      atualizacao das condicoes
8  }
```



# Estrutura de Repetição: WHILE

## Vamos praticar

```
1 #include <stdio.h>
2
3 int main( ){
4
5     int num;
6     scanf("%d", &num);
7
8     while( num > 0 )
9     {
10        printf("%d\n", num/2);
11        scanf("%d", &num);
12    }
13
14    return 0;
15 }
16
```

# Estrutura de Repetição: WHILE

## Estrutura Básica - Múltiplas condições e atualizações

```
1 //declaracoes
2 int i, j;
3
4 //inicializacoes
5 j=3;
6 scanf("%d", &i);
7
8 //repeticao com multiplas condicoes
9 while ( ( i != 0 ) && ( j > 0 ) )
10 {
11     ...
12     //atualizacoes
13     j--;
14     scanf("%d", &i);
15 }
16
```

## Estrutura de Repetição: WHILE

### Vamos praticar

Faça um programa que leia números até que a soma de todos os números digitados for pelo menos 20. Imprimir a soma final.

```
1
2  inicializacao das condicoes
3
4  while ( condicoes )
5  {
6      ...
7      atualizacao das condicoes
8  }
```

# Estrutura de Repetição: WHILE

## Vamos praticar

```
1 #include <stdio.h>
2
3 int main( ){
4
5     //nao esqueca de inicializar a variavel
6     //zero: elemento neutro da soma
7     int total = 0;
8     int num;
9
10    while( total < 20 )
11    {
12        printf("Total = %d\n", total);
13        scanf("%d", &num);
14
15        total = total + num;
16    }
17
18    printf("Final total = %d\n", total);
19    return 0;
20 }
21
```

## Estrutura de Repetição: WHILE

### Vamos praticar

Leia um número, calcule o fatorial desse número e imprima na tela

$$0! = 1$$

$$1! = 1$$

$$2! = 2$$

$$n! = n*(n-1)!$$

$$n! = n*(n-1)*(n-2)*(n-3)*...*3*2*1$$

```
1
2  inicializacao das condicoes
3
4  while ( condicoes )
5  {
6      ...
7      atualizacao das condicoes
8  }
```

# Estrutura de Repetição: WHILE

## Vamos praticar

```
1 //n! = n*(n-1)*(n-2)*(n-3)*...*3*2*1
2 //declarar
3 int fat, n, i;
4
5 //inicializar
6 scanf("%d", &n);
7 fat = 1;
8 i = n;
9
10 //repetir - verificar
11 while(n >= 1)
12 {
13     fat = fat * n;
14
15     //atualizar
16     n--;
17 }
18
19 printf("%d! = %d\n", i, fat);
20
21
```

## Estrutura de Repetição: WHILE

### Vamos praticar

Leia um número, calcule a sequência de Fibonacci desse número e imprima na tela

$$F(0) = 0$$

$$F(1) = 1$$

$$F(2) = F(1) + F(0) = 1 + 0 = 1$$

$$F(3) = F(2) + F(1) = 1 + 1 = 2$$

$$F(4) = F(3) + F(2) = 2 + 1 = 3$$

...

$$F(n) = F(n-1) + F(n-2)$$

# Estrutura de Repetição: WHILE

## Vamos praticar

```
1 //declarar
2 int n, f_ant, f_ant_ant, f_atual;
3
4 //inicializar
5 scanf("%d", &n);
6
7 f_ant_ant = 0; //F(0)
8 f_ant = 1; //F(1)
9 f_atual = n;
10
11 //repetir de F(2)=F(1)+F(0) ate F(n)=F(n-1)+F(n-2)
12 while(n >= 2)
13 {
14     f_atual = f_ant + f_ant_ant;
15     f_ant_ant = f_ant;
16     f_ant = f_atual;
17     n--;
18 }
19
20 //F(n)
21 printf("F(%d) = %d\n", n, f_atual);
22
23
```



# Estrutura de Repetição: WHILE

## Observação!

```
1 //leituras sucessivas de caracteres
2 char a, b;
3 scanf("%c", &a);
4 scanf("%c", &b);
5 printf("%c %c", a, b); //saida??
6
7 scanf("%c", &a);
8 scanf(" %c", &b); //%c é literal! vai ler um caractere
9     ^           //caso queria que leia o próximo caractere,
10    |           //tem que colocar um espaço antes pois assim
11           //o scanf vai pular para a próxima leitura
12 printf("%c %c", a, b); //saida??
13
```

- 1 Estruturas de Repetição
- 2 Estruturas de Repetição: WHILE
- 3 Condições (if) + Repetições (while)
  - Repetição dentro da condição
  - Condição dentro de repetição
  - Repetição dentro de repetição
- 4 Estrutura de Repetição: FOR

- 1 Estruturas de Repetição
- 2 Estruturas de Repetição: WHILE
- 3 Condições (if) + Repetições (while)
  - Repetição dentro da condição
  - Condição dentro de repetição
  - Repetição dentro de repetição
- 4 Estrutura de Repetição: FOR

# Estrutura de Repetição: WHILE

## Condições + Repetições

### Repetição dentro da condição

```
1  if ( )  
2  {  
3      while ( )  
4      {  
5          ...  
6      }  
7  }  
8
```

# Estrutura de Repetição: WHILE

## Repetição dentro da condição

Exemplo: Dado um número, dizer se é primo (divisível somente por 1 e por ele mesmo).

```
1  int a, i;
2  scanf("%d", &a);
3  a = abs(a);
4
5  //procurando por um divisor exato impar
6  //de 3 ate a
7  i = 3;
8  while ( i<a && a%i!=0 ) {
9      i=i+2;
10 }
11
12 if ( a==2 || i==a ) {
13     printf("eh primo\n");
14 } else {
15     printf("nao eh primo\n");
16 }
17
```

# Estrutura de Repetição: WHILE

## Repetição dentro da condição

Exemplo: Dado um número, dizer se é primo (divisível somente por 1 e por ele mesmo).

```
1  int a, i;
2  scanf("%d", &a);
3  a = abs(a);
4
5  //procurando por um divisor exato impar
6  //de 3 ate a
7  i = 3;
8  while ( i<a && a%i!=0 ) {
9      i=i+2;
10 }
11
12 if ( a==2 || i==a ) {
13     printf("eh primo\n");
14 } else {
15     printf("nao eh primo\n");
16 }
17
```

E se o número for par? Precisa procurar o divisor?  
2 é primo e os outros pares não são primos.

# Estrutura de Repetição: WHILE

## Repetição dentro da condição

```
1  int a, i;
2
3  scanf("%d", &a);
4  a = abs(a);
5  i = 3;
6
7  if(a%2!=0)
8  {
9      //procurando por um divisor exato
10     while( i<a && a%i!=0)
11     {
12         i=i+2;
13     }
14 }
15
16 if ( a==2 || i==a )
17 {
18     printf("eh primo\n");
19 }
20 else
21 {
22     printf("nao eh primo\n");
23 }
24
```

- 1 Estruturas de Repetição
- 2 Estruturas de Repetição: WHILE
- 3 Condições (if) + Repetições (while)
  - Repetição dentro da condição
  - **Condição dentro de repetição**
  - Repetição dentro de repetição
- 4 Estrutura de Repetição: FOR



# Estrutura de Repetição: WHILE

## Condições + Repetições

Repetição dentro da condição

```
1  if( )  
2  {  
3      while( ){ }  
4  }  
5
```

Condição dentro de repetição

```
1  while( )  
2  {  
3      if( )  
4      {  
5          ...  
6      }  
7  }  
8
```

# Estrutura de Repetição: WHILE

## Condição dentro da repetição

Exemplo: Número primo (versão menos eficiente)

```
1  int a, i, d;
2
3  scanf("%d", &a);
4  a = abs(a);
5  i = 1;
6  d = 0;
7
8  while( i<=a ) {
9      if(a%i==0){
10         d++;
11     }
12     i++;
13 }
14
15 if ( d==2 )
16 {
17     printf("eh primo\n");
18 }
19 else
20 {
21     printf("nao eh primo\n");
22 }
23
```

# Estrutura de Repetição: WHILE

## Condição dentro da repetição

Exemplo: Ler 10 números e somar se for par

```
1  int n;  
2  int x;  
3  int soma = 0;  
4  
5  n=0;  
6  while( n<10 )  
7  {  
8      scanf("%d", &x);  
9      if( x%2 == 0 )  
10     {  
11         soma = soma + x;  
12     }  
13     n++;  
14 }  
15 printf("Soma %d\n", soma);  
16
```

# Estrutura de Repetição: WHILE

## Condição dentro de repetição

Exemplo: Ler números, somar os números pares e parar quando ler o número zero ou a soma for pelo menos 20.

```
1  int total = 0;
2  int num;
3
4  scanf("%d", &num);
5  while(num!=0 || total<20)
6  {
7      if(num%2==0)
8      {
9          total = total + num;
10     }
11     scanf("%d", &num);
12 }
13 printf("Total = %d\n", total);
14
```

## Vamos praticar

URI Online Judge | 1059

### Números Pares

Adaptado por Neilor Tonin, URI  Brasil

**Timelimit: 1**

Faça um programa que mostre os números pares entre 1 e 100, inclusive.

#### Entrada

Neste problema extremamente simples de repetição não há entrada.

#### Saída

Imprima todos os números pares entre 1 e 100, inclusive se for o caso, um em cada linha.

#### Exemplo de Entrada

#### Exemplo de Saída

```
2
4
6
...
100
```

# Estrutura de Repetição: WHILE

## Vamos praticar - versão 1

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int i;
6
7     //inicializar condicao - variavel de controle
8     i=1;
9
10    //repeticao - enquanto i for menor ou igual a 100
11    while(i <= 100)
12    {
13        if(i%2==0)
14            printf("%d\n", i);
15        i++;
16    }
17    return 0;
18 }
19
```

# Estrutura de Repetição: WHILE

## Vamos praticar - versão 2

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int i;
6
7     //inicializar condicao - variavel de controle
8     i=2;
9
10    //repeticao - enquanto i for menor ou igual a 100
11    while(i <= 100)
12    {
13        printf("%d\n", i);
14
15        //percorre os numeros pares
16        i+=2;
17    }
18    return 0;
19 }
20
```

# Estrutura de Repetição: WHILE

## Vamos praticar

URI Online Judge | 1060

## Números Positivos

Adaptado por Neilor Tonin, URI  Brasil

**Timelimit: 1**

Faça um programa que leia 6 valores. Estes valores serão somente negativos ou positivos (desconsidere os valores nulos). A seguir, mostre a quantidade de valores positivos digitados.

### Entrada

Seis valores, negativos e/ou positivos.

### Saída

Imprima uma mensagem dizendo quantos valores positivos foram lidos.

#### Exemplo de Entrada

```
7
-5
6
-3.4
4.6
12
```

#### Exemplo de Saída

```
4 valores positivos
```



# Estrutura de Repetição: WHILE

## Vamos praticar

```
1 #include <stdio.h>
2
3 int main()
4 {
5     //numero a ser lido
6     float num;
7
8     //declarar e inicializar
9     int i=1, qtd_positivos=0;
10
11    //repetir - verificar
12    while(i<=6)
13    {
14        scanf("%f", &num);
15        if(num>=0)
16            qtd_positivos = qtd_positivos+1;
17
18        //atualizar
19        i = i+1;
20    }
21
22    printf("%d valores positivos\n", qtd_positivos);
23    return 0;
24 }
25
26
```

# Estrutura de Repetição: WHILE

## Vamos praticar

URI Online Judge | 1071

### Soma de Impares Consecutivos I

Adaptado por Neilor Tonin, URI  Brasil

**Timelimit:** 1

Leia 2 valores inteiros **X** e **Y**. A seguir, calcule e mostre a soma dos números ímpares entre eles.

#### Entrada

O arquivo de entrada contém dois valores inteiros.

#### Saída

O programa deve imprimir um valor inteiro. Este valor é a soma dos valores ímpares que estão entre os valores fornecidos na entrada que deverá caber em um inteiro.

Exemplo de Entrada	Exemplo de Saída
6 -5	5
15 12	13
12 12	0

# Estrutura de Repetição: WHILE

## Vamos praticar

```
1  int x, y, soma;
2
3  scanf("%d %d", &x, &y);
4  soma = 0;
5
6  if(x <= y)
7  {
8      x++;
9      while(x < y)
10     {
11         if(x%2!=0) soma += x;
12         x++;
13     }
14 }
15 else
16 {
17     y++;
18     while(y < x)
19     {
20         if(y%2!=0) soma += y;
21         y++;
22     }
23 }
24 printf("%d\n", soma);
25
26
```

# Estrutura de Repetição: WHILE


## Vamos praticar

```
1  int x, y, soma;
2
3  scanf("%d %d", &x, &y);
4  soma = 0;
5
6  if(x <= y)
7  {
8      while(++x < y)
9      {
10         if(x%2!=0) soma += x;
11     }
12 }
13 else
14 {
15     while(++y < x)
16     {
17         if(y%2!=0) soma += y;
18     }
19 }
20 printf("%d\n", soma);
21
22
```

## Vamos praticar

URI Online Judge | 1064

### Positivos e Média

Adaptado por Neilor Tonin, URI  Brasil

**Timelimit: 1**

Leia 6 valores. Em seguida, mostre quantos destes valores digitados foram positivos. Na próxima linha, deve-se mostrar a média de todos os valores positivos digitados, com um dígito após o ponto decimal.

#### Entrada

A entrada contém 6 números que podem ser valores inteiros ou de ponto flutuante. Pelo menos um destes número será positivo.

#### Saída

O primeiro valor de saída é a quantidade de valores positivos. A próxima linha deve mostrar a média dos valores positivos digitados.

#### Exemplo de Entrada

```
7
-5
6
-3.4
4.6
12
```

#### Exemplo de Saída

```
4 valores positivos
7.4
```

# Estrutura de Repetição: WHILE

## Vamos praticar

```
1  float num, soma, media;
2  int qtd_positivos, i;
3
4  qtd_positivos=0;
5  soma=0.0;
6
7  scanf("%f", &num);
8
9  i=1;
10 while(i<=6)
11 {
12     scanf("%f", &num);
13     if(num>=0)
14     {
15         qtd_positivos++;
16         soma+=num;
17     }
18     i++;
19 }
20
21 media = soma/qtd_positivos;
22
23 printf("%d valores positivos\n", qtd_positivos);
24 printf("%.1f\n", media);
25
26
```

- 1 Estruturas de Repetição
- 2 Estruturas de Repetição: WHILE
- 3 Condições (if) + Repetições (while)
  - Repetição dentro da condição
  - Condição dentro de repetição
  - Repetição dentro de repetição
- 4 Estrutura de Repetição: FOR

# Estrutura de Repetição: WHILE

## Condições + Repetições

### Repetição dentro da condição

```
1  if ( )  
2  {  
3      while ( ){ }  
4  }  
5
```

### Condição dentro de repetição

```
1  while ( )  
2  {  
3      if ( ){ }  
4  }  
5
```

### Repetição dentro de repetição

```
1  while ( )  
2  {  
3      while ( ){ }  
4  }  
5
```



# Estrutura de Repetição: WHILE

## Repetição dentro da repetição

Exemplo: Imprimir as posições de uma matriz 4x5

```
1  int linhas;  
2  int colunas;  
3  
4  while( )  
5  {  
6      while( )  
7      {  
8      }  
9  }  
10
```

# Estrutura de Repetição: WHILE

## Repetição dentro da repetição

Exemplo: Imprimir as posições de uma matriz 4x5

```
1  int linhas;  
2  int colunas;  
3  
4  linhas=0;  
5  while( linhas < 4)  
6  {  
7      while( )  
8      {  
9      }  
10  
11     linhas++;  
12 }  
13
```

# Estrutura de Repetição: WHILE

## Repetição dentro da repetição

Exemplo: Imprimir as posições de uma matriz 4x5

```
1  int linhas;  
2  int colunas;  
3  
4  linhas=0;  
5  while( linhas < 4)  
6  {  
7      colunas=0;  
8      while( colunas < 5)  
9      {  
10         colunas++;  
11     }  
12  
13     linhas++;  
14 }  
15
```

# Estrutura de Repetição: WHILE

## Repetição dentro da repetição

Exemplo: Imprimir as posições de uma matriz 4x5

```
1  int linhas;  
2  int colunas;  
3  
4  linhas=0;  
5  while( linhas < 4)  
6  {  
7      colunas=0;  
8      while( colunas < 5)  
9      {  
10         printf( "[%d,%d]", linhas, colunas);  
11         colunas++;  
12     }  
13     printf( "\n");  
14     linhas++;  
15 }  
16
```

## Estrutura de Repetição: WHILE

### Repetição dentro da repetição

Exemplo: Imprimir as posições de uma matriz 4x5

```
[0,0][0,1][0,2][0,3][0,4]  
[1,0][1,1][1,2][1,3][1,4]  
[2,0][2,1][2,2][2,3][2,4]  
[3,0][3,1][3,2][3,3][3,4]
```

# Estrutura de Repetição: WHILE

## Repetição dentro da repetição

Exemplo: Vários TESTES, com vários casos para cada TESTE

```
1  int testes, caso;
2
3  //quantidade de testes
4  scanf("%d", &testes);
5  while( testes != 0 )
6  {
7
8      //para cada teste
9      while( testes > 0)
10     {
11         scanf("%d", &caso);
12         //processa um caso
13         testes--;
14     }
15
16     //proxima bateria de testes
17     scanf("%d", &testes);
18 }
19
```

# Estrutura de Repetição: WHILE

## Repetição dentro da repetição

Exemplo: Vários TESTES, com vários casos para cada TESTE

```
1  int testes, caso;
2  int num_t;
3
4  scanf("%d", &testes);
5  //contador de testes
6  num_t=1;
7  while( testes != 0 )
8  {
9      //numero do teste (1, 2, 3, ...)
10     printf("Teste %d\n", num_t);
11     while( testes > 0)
12     {
13         scanf("%d", &caso);
14         //processa tarefa
15         testes--;
16     }
17
18     scanf("%d", &testes);
19     num_t++;
20 }
21
```

# Estrutura de Repetição: WHILE

## Repetição dentro da repetição

Exemplo: Vários TESTES, com vários casos para cada TESTE

```
1  int testes, caso;
2  int num_t, num_c;
3
4  scanf("%d", &testes);
5  num_t=1;
6  while( testes != 0 ) {
7      printf("Teste %d\n", num_t++);
8
9      //contador de casos
10     num_c=1;
11     while( testes > 0) {
12         //numero do caso
13         printf("Caso %d\n", num_c++);
14
15         scanf("%d", &caso);
16         //processa tarefa
17
18         testes--;
19     }
20     scanf("%d", &testes);
21 }
```



# Estrutura de Repetição: WHILE

## Vamos praticar

URI Online Judge | 1096

## Sequencia IJ 2

Adaptado por Neilor Tonin, URI  Brasil

**Timelimit: 1**

Você deve fazer um programa que apresente a sequencia conforme o exemplo abaixo.

### Entrada

Não há nenhuma entrada neste problema.

### Saída

Imprima a sequencia conforme exemplo abaixo

#### Exemplo de Entrada

#### Exemplo de Saída

```
I=1 J=7
I=1 J=6
I=1 J=5
I=3 J=7
I=3 J=6
I=3 J=5
...
I=9 J=7
I=9 J=6
I=9 J=5
```

# Estrutura de Repetição: WHILE

## Vamos praticar

```
1 #include <stdio.h>
2
3 int main( ){
4
5     int i, j;
6
7     i = 1;
8     while (i <= 9)
9     {
10        j = 7;
11        while (j >= 5)
12        {
13            printf("I=%d J=%d\n", i, j);
14            j--;
15        }
16        i+=2;
17    }
18    return 0;
19 }
20
```

## Estrutura de Repetição: WHILE

### Vamos praticar

Imprima a tabuada do 2 ao 9. Identificando a tabua com a mensagem "Tabuada do"e separando as tabuadas por uma linha em branco.

Saída:

Tabuada do 2

2x0=0

2x1=2

...

2x10=20

Tabuada do 3

3x0=0

3x1=3

...

# Estrutura de Repetição: WHILE

## Vamos praticar

```
1 #include <stdio.h>
2
3 int main( ){
4
5     int i, j;
6
7     i = 2;
8     while (i <= 9)
9     {
10        printf("Tabuado do %d\n", i);
11        j = 0;
12        while (j <= 10)
13        {
14            printf("%dx%d=%d\n", i, j, i*j);
15            j++;
16        }
17        printf("\n");
18        i++;
19    }
20    return 0;
21 }
22
```

## Vamos praticar

URI Online Judge | 1099

## Soma de Ímpares Consecutivos II

Adaptado por Neilor Tonin, URI  Brasil

**Timelimit: 1**

Leia um valor inteiro **N** que é a quantidade de casos de teste que vem a seguir. Cada caso de teste consiste de dois inteiros **X** e **Y**. Você deve apresentar a soma de todos os ímpares existentes *entre X e Y*.

### Entrada

A primeira linha de entrada é um inteiro **N** que é a quantidade de casos de teste que vem a seguir. Cada caso de teste consiste em uma linha contendo dois inteiros **X** e **Y**.

### Saída

Imprima a soma de todos valores ímpares *entre X e Y*.

#### Exemplo de Entrada

```
7
4 5
13 10
6 4
3 3
3 5
3 4
3 8
```

#### Exemplo de Saída

```
0
11
5
0
0
0
12
```

# Estrutura de Repetição: WHILE

## Vamos praticar

```
1  int n, x, y, s;
2
3  scanf("%d", &n);
4  while(n>0)
5  {
6      scanf("%d%d", &x, &y);
7
8      s=0;
9      if(x <= y)
10     {
11         while(++x < y)
12             if(x%2!=0) s += x;
13     }
14     else
15     {
16         while(++y < x)
17             if(y%2!=0) s += y;
18     }
19     printf("%d\n", s);
20
21     n--;
22 }
23
24
```

# Estrutura de Repetição: WHILE

## Relembrando a Estrutura Básica - Geral

```
1  inicializacao das condicoes: atribuicao ou entrada
2
3  while ( condicoes )
4  {
5      instrucoes
6      atualizacao das condicoes: atribuicao ou entrada
7  }
```

- 1 Estruturas de Repetição
- 2 Estruturas de Repetição: WHILE
- 3 Condições (if) + Repetições (while)
  - Repetição dentro da condição
  - Condição dentro de repetição
  - Repetição dentro de repetição
- 4 Estrutura de Repetição: FOR



# Estrutura de Repetição: FOR

## Estrutura Básica - Geral

```
1  for ( inicializacoes; condicoes; atualizacoes )
2  {
3      instrucoes
4  }
5
6
```

## Estrutura de Repetição: FOR

### Exemplo: percorrer linhas e colunas

```
1  int linhas;  
2  int colunas;  
3  
4  for(linhas=0; linhas < 4; linhas++)  
5  {  
6      for(colunas=0; colunas < 5; colunas++)  
7      {  
8          printf("[%d,%d]", linhas, colunas);  
9      }  
10     printf("\n");  
11 }  
12
```