

# Estruturas de controle de fluxo

- Instruções de um programa não seguem necessariamente um fluxo linear de começo, meio e fim;
- Bifurcações, laços etc. → Lógica

# Estruturas de controle de fluxo

- Instruções condicionais – *if else*
  - if (condicao1) instrucao

```
#include <stdio.h>

void main()
{
    int x = 10;

    if (x>0) printf("x eh positivo");
}
```

# Estruturas de controle de fluxo

- Instruções condicionais – *if else*
  - if (condicao1) instrucao

```
#include <stdio.h>

void main()
{
    int x = 10;

    if (x>0)
        printf("x eh positivo");
}
```

```
#include <stdio.h>

void main()
{
    int x = 10;

    if (x>0)
    {
        printf("x eh positivo");
    }
}
```

# Estruturas de controle de fluxo

- Instruções condicionais – *if else*

if (condicao1) instrucao1 else instrucao2

```
#include <stdio.h>

void main()
{
    int x = 10;

    if (x>0) printf("x eh positivo");
    else printf("x eh negativo");
}
```

# Estruturas de controle de fluxo

- Instruções condicionais – *if else* concatenados

```
if (condicao1) instrucao1  
else if(condicao2) instrucao2
```

```
...
```

```
else instrucaoN
```

# Estruturas de controle de fluxo

- Instruções condicionais – *if else* concatenados

```
#include <stdio.h>

void main()
{
    int x = 10;

    if (x>0) printf("x eh positivo: %d",x);
    else if (x==0) printf("x eh igual a zero");
    else printf("x eh negativo: %d",x);
}
```

# Estruturas de controle de fluxo

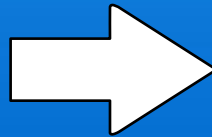
- Instruções iterativas – *while*
  - *while(condicao) instrucao*

```
#include <stdio.h>
void main()
{
    int x = 10;
    while(x>0)
    {
        printf("%d ",x);
        x--;
    }
    printf("BOOM!!!!");
}
```

# Estruturas de controle de fluxo

- Instruções iterativas – *while*
  - *while(condicao) instrucao*

```
#include <stdio.h>
void main()
{
    int x = 10;
    while(x>0)
    {
        printf("%d ",x);
        x--;
    }
    printf("BOOM!!!!");
}
```



10 9 8 7 6 5 4 3 2 1 BOOM!!!!



# Estruturas de controle de fluxo

- Instruções iterativas – *while*
  - *while(condicao) instrucao*

```
#include <stdio.h>
void main()
{
    int x = 10;
    while(x>0)
    {
        printf("%d ",x);
        x--;
    }
    printf("BOOM!!!!");
}
```

```
#include <stdio.h>

void main()
{
    int x = 10;
    while(x>0) printf("%d ",x--);
    printf("BOOM!!!!");
}
```

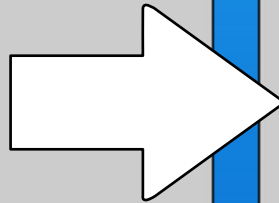
# Estruturas de controle de fluxo

- Instruções iterativas – *while*
  - *while(condicao) instrucao*

```
#include <stdio.h>

void main()
{
    unsigned char x = 10;

    while(x>=0) printf("%d ", x--);
    printf("BOOM!!!!");
}
```



10 9 8 7 6 5 4 3 2 1 0 255 254 253  
252 251 250 249 248 247 246 245  
244 243 242 241 240 239 238 237  
236 235 234 233 232 231 230 .....

x vale:

10 → 9 → 8 → 7 → 6 → 5 →  
4 → 3 → 2 → 1 → 0 → 255  
por ser *unsigned char*

# Estruturas de controle de fluxo

- Instruções iterativas – *while*
  - *while(condicao) instrucao*

```
#include <stdio.h>

void main()
{
    int x = 10;

    while(x>0) printf("%d ",x);
    printf("BOOM!!!!");
}
```

LOOP INFINITO!!!!

# Estruturas de controle de fluxo

- Instruções iterativas – *while*
  - *while(condicao) instrucao*

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

```
void main()
```

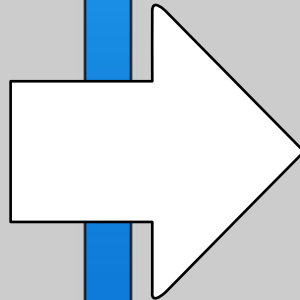
```
{
```

```
    unsigned char x = 10;
```

```
    while(x>10) printf("%d ",x--);
```

```
    printf("BOOM!!!!");
```

```
}
```



*BOOM!!!!*

O programa não entra  
na instrução *while()*

# Estruturas de controle de fluxo

- Instruções iterativas – *do while*
  - *do instrucao while(condicao);*

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

```
void main()
```

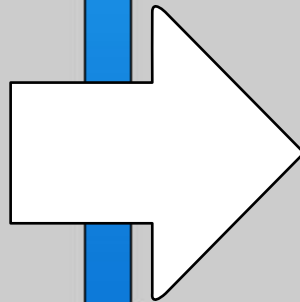
```
{
```

```
    unsigned char x = 10;
```

```
    do printf("%d ",x--); while(x>10);
```

```
    printf("BOOM!!!!");
```

```
}
```



*10 BOOM!!!!*

O programa realiza o que a instrução *do* manda e sai no *while()*

# Estruturas de controle de fluxo

- Instruções iterativas – *do while*
  - *do instrucao while(condicao);*

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

```
void main()
```

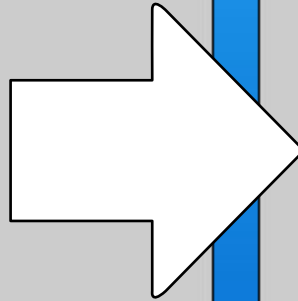
```
{
```

```
    unsigned char x = 10;
```

```
    do printf("%d ",x--); while(x>0);
```

```
    printf("BOOM!!!!");
```

```
}
```



10 9 8 7 6 5 4 3 2 1 BOOM!!!!

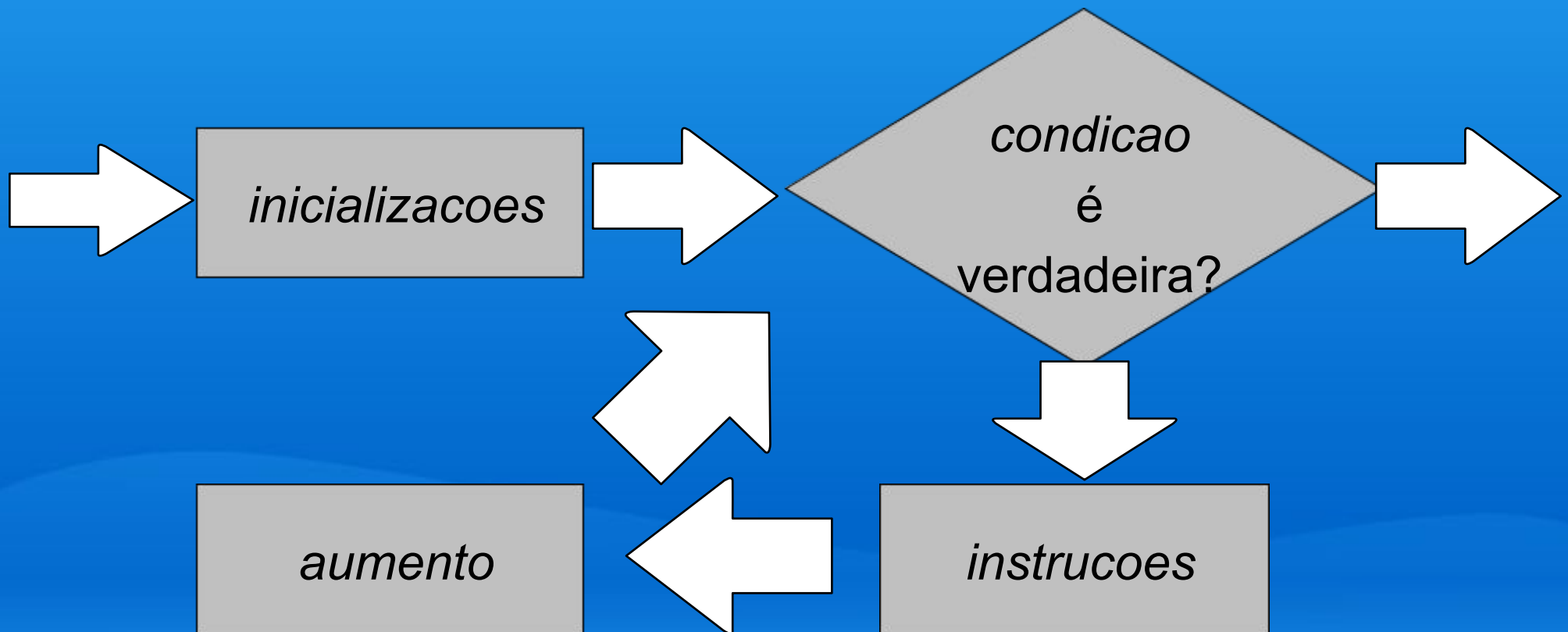
O programa realiza o que a instrução *do* comanda enquanto a expressão dentro do *while()* é verdadeira

# Estruturas de controle de fluxo

- Instruções iterativas – *for*
  - *for(inicializacoes; condicao; aumento) instrucoes*
- Repete *instrucoes* enquanto *condicao* é verdadeira;
- Oferece campos especiais para inicializar e incrementar variáveis.

# Estruturas de controle de fluxo

- Instruções iterativas – *for*
  - *for(inicializacoes; condicao; aumento) instrucoes*





# Estruturas de controle de fluxo

- Instruções iterativas – *for*
  - *for(inicializacoes; condicao; aumento) instrucoes*

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

```
void main()
```

```
{
```

```
    unsigned char x;
```

```
    for (x=10; x>0; x--) printf("%d ",x);
```

```
    printf("BOOM!!!!");
```

```
}
```



10 9 8 7 6 5 4 3 2 1 BOOM!!!!

# Estruturas de controle de fluxo

- Instruções iterativas – *for*

- *for(inicializacoes; condicao; aumento) instrucoes*

```
#include <stdio.h>

void main()
{
    unsigned char x, y;

    for (x=10, y = 0; x!=y; x--, y++)
        printf("(%d,%d) ",x,y);
}
```



(10,0) (9,1) (8,2) (7,3) (6,4)

# Estruturas de controle de fluxo

- Instruções de salto – *break*
  - Interrompe um *loop* antes da hora

```
#include <stdio.h>
void main()
{
    unsigned char x;
    for (x=10; x>0; x--)
    {
        if(x==5) break;
        printf("%d ",x);
    }
    printf("BOOM!!!!");
}
```

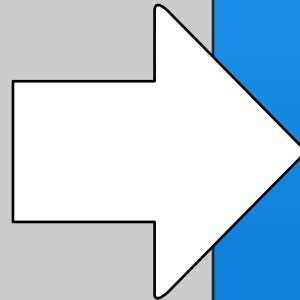


10 9 8 7 6 BOOM!!!!

# Estruturas de controle de fluxo

```
#include <stdio.h>

void main()
{
    unsigned char x, y;
    for (x=0; x<3; x++)
    {
        printf("x = %d, y = ",x);
        for (y=0;y<3;y++)
        {
            if(x==1) break;
            printf("%d ",y);
        }
        printf("\n");
    }
}
```

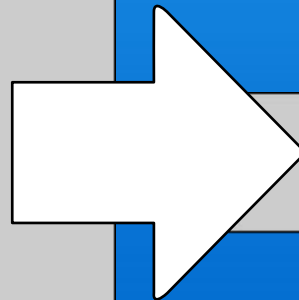


```
x = 0, y = 0 1 2
x = 1, y =
x = 2, y = 0 1 2
```

# Estruturas de controle de fluxo

- Instruções de salto – *goto*
  - Salto incondicional
  - Resquício de outras linguagens – não use!!

```
#include <stdio.h>
void main()
{
    int x=10;
    loop:
    printf("%d ",x--);
    if(x>0) goto loop;
    printf("BOOM!!!!");
}
```



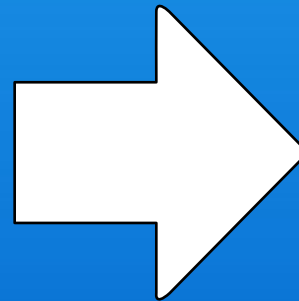
10 9 8 7 6 5 4 3 2 1 BOOM!!!!

# Estruturas de controle de fluxo

- Instruções de seleção – *switch*
  - Semelhante a *ifs* e *elses* concatenados

# Estruturas de controle de fluxo

```
#include <stdio.h>
void main()
{
    int x=1;
    switch(x)
    {
        case 1:
            printf("x = 1");
            break;
        case 2:
            printf("x = 2");
            break;
        default:
            printf("NDA");
    }
}
```



*x = 1*

# Estruturas de controle de fluxo

```
#include <stdio.h>
void main()
{
    int x=1;
    switch(x)
    {
        case 1:
            printf("x = 1");
            break;
        case 2:
            printf("x = 2");
            break;
        default:
            printf("NDA");
    }
}
```

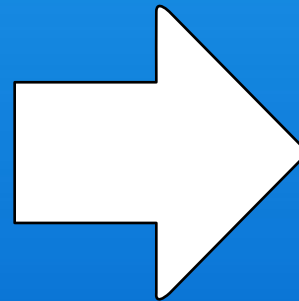
```
#include <stdio.h>

void main()
{
    int x=1;
    if (x==1) printf("x = 1");
    else if (x==2) printf("x = 2");
    else printf("NDA");
}
```



# Estruturas de controle de fluxo

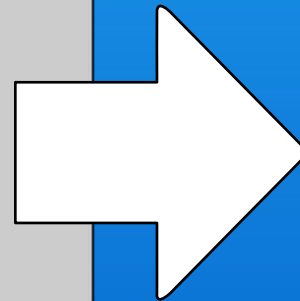
```
#include <stdio.h>
void main()
{
    int x=40;
    switch(x)
    {
        case 1:
            printf("x = 1");
            break;
        case 2:
            printf("x = 2");
            break;
        default:
            printf("NDA");
    }
}
```



*NDA*

# Estruturas de controle de fluxo

```
#include <stdio.h>
void main()
{
    int x=1;
    switch(x)
    {
        case 1:
        case 2:
        case 3:
            printf("x = 1, 2 ou 3");
            break;
        case 4:
            printf("x = 4");
            break;
        default:
            printf("NDA");
    }
}
```



*x = 1, 2 ou 3*