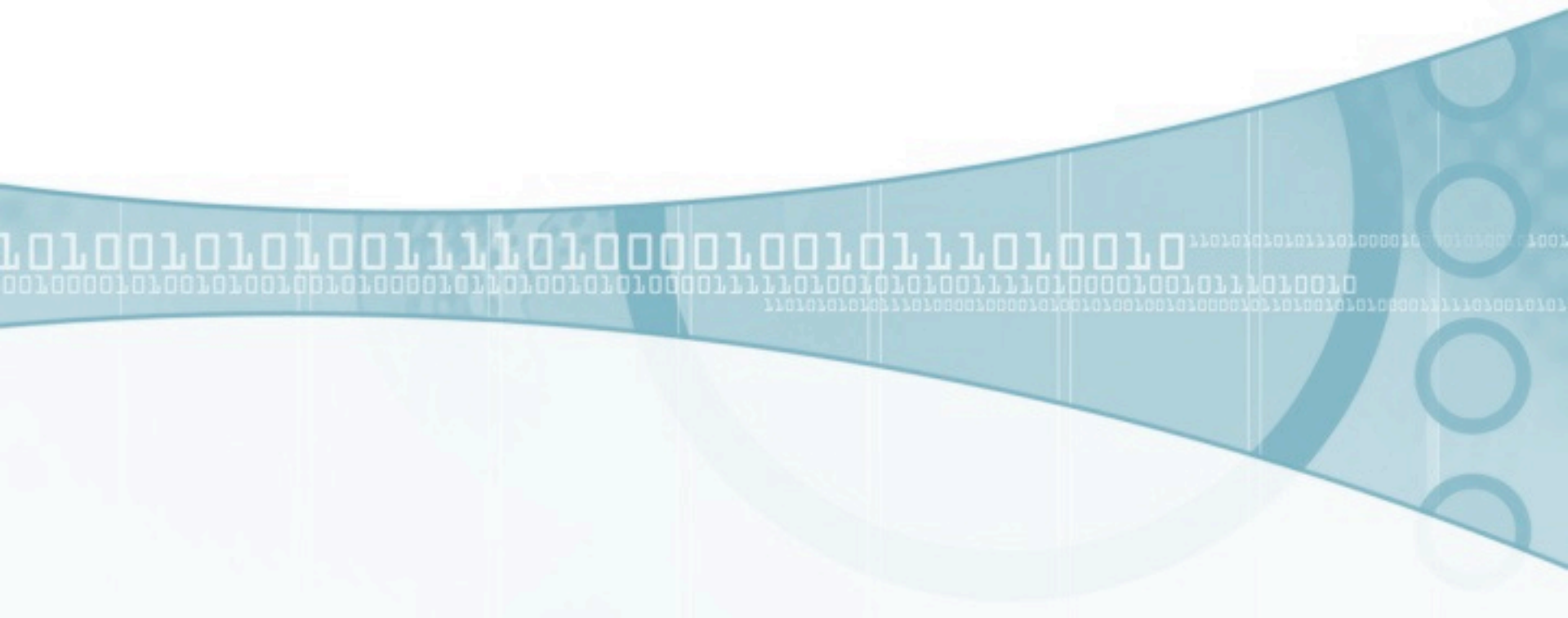


Structures algorithmiques



Boucle *tant_que*

```
tantque <cond> faire  
    corpstantque;  
fintantque
```



```
tantque:  
    si <!  
cond> alors  
        goto fintantque;  
    corpstantque;  
    goto tantque;  
fintantque: ...
```

```
tantque i > j faire  
    i = i-1;
```



```
tantque:  
    cmp r1,r2  
    bls fintantque  
    sub r1,r1,#1  
    b tantque  
fintantque:  
    ...
```

Boucle *pour*

```
pour i allant de 1 à N par pas de 1  
  corpsboucle;
```



```
i=1;  
tantque i <= N faire  
  corpsboucle;  
  i=i+1;  
fintantque
```



*voir la diapo
précédente !*

Boucles avancées

tantque:

```
cmp r1,#0
beq fintantque
sub r1,r1,#1
b tantque
```

fintantque:

...



```
cmp r1,#0
```

tantque:

```
beq fintantque
sub s r1,r1,#1
b tantque
```

fintantque:

...

```
cmp r1,#0
```

beq fintantque

tantque:

```
sub s r1,r1,#1
bne tantque
```

fintantque:

...



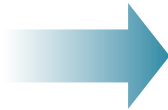
Conditionnelle *si...alors...*

```
si <cond> alors  
    corpssi;  
finsi
```



```
si:  
    si <!cond> alors  
        goto finssi;  
    corpssi;  
finsi: ...
```

```
si a==0 alors  
    a = a+1;  
finsi  
b = a;
```



```
si:  
    cmp r1,#0  
    bne finssi  
    add r1,r1,#1  
finsi:  
    mov r2,r1
```



Conditionnelle *si...alors...sinon...*

```
si <cond> alors  
    corpsalors;  
sinon  
    corpsinon;  
finsi
```



```
si:  
    si <!  
    goto sinon;  
    corpsalors;  
    goto fin  
sinon:  
    corpssinon;  
finsi: ...
```

```
si a==0 alors  
    a = a+1;  
sinon  
    a = a+2;  
finsi
```



```
si: cmp r1, #0  
    bne sinon  
    add r1, r1, #1  
    b fin  
sinon:  
    add r1, r1, #2  
finsi:
```

