

Algorithmique et programmation

M. ROUAN-SERIK

Institut de maintenance et de sécurité industrielle IMSI
Université d'Oran 2 Mohamed Ben Ahmed.
mehdi.rouan@gmail.com



Plan

Introduction

Structures de contrôle

Structures itératives

Conclusion

Notions d'algorithmique

- ▶ Une suite finie et non ambiguë d'opérations permettant de résoudre un problème
- ▶ Il tire son nom du savant musulman *Al-Khwârizmî*.
- ▶ Algorithme d'*Euclide*, crible d'*Ératosthène*, etc. . .
- ▶ Usage quotidien : fonctionnement des ordinateurs, cryptographie, routage d'information, traitement d'image, etc. . .

Algorithme simple

Un exemple typique d'algorithme qu'on utilise parfois tous les jours, il s'agit de l'échange entre deux variables.

Algorithme : Echange ;

var x, y, z : réel ;

Début

lire(x);

lire(y);

z \leftarrow y ;

y \leftarrow x ;

x \leftarrow z ;

ecrire("x = ", x, "y = ", y);

Fin.

Plan

Introduction

Structures de contrôle

Structures itératives

Conclusion

Conditions

Syntaxe en Algorithmique

Si *Conditions* **alors**

 insts;

Sinon

 insts';

Finsi;

Syntaxe en Matlab

```
if condition
insts ;
else
insts ' ;
end
```

Conditions II

Parité d'un entier

Dites si un nombre $n \in \mathbb{N}$ est pair ou impair !

Solution

$$n : \begin{cases} \text{pair,} & \text{si } n = 2k \\ \text{impair,} & \text{sinon} \end{cases}$$

```
n=input('n = ');  
if mod(n,2) == 0  
disp('pair');  
else  
disp('impair');  
end
```

Conditions II

Parité d'un entier

Dites si un nombre $n \in \mathbb{N}$ est pair ou impair !

Solution

$$n : \begin{cases} \text{pair,} & \text{si } n = 2k \\ \text{impair,} & \text{sinon} \end{cases}$$

```
n=input('n = ');  
if mod(n,2) == 0  
disp('pair');  
else  
disp('impair');  
end
```

Conditions II

Parité d'un entier

Dites si un nombre $n \in \mathbb{N}$ est pair ou impair !

Soluion

$$n : \begin{cases} \text{pair,} & \text{si } n = 2k \\ \text{impair,} & \text{sinon} \end{cases}$$

```
n=input('n = ');  
if mod(n,2) == 0  
disp('pair');  
else  
disp('imampir');  
end
```

Plan

Introduction

Structures de contrôle

Structures itératives

Conclusion

Boucle pour

Syntaxe en Algorithmique

pour (*i de 1 à 10*, 1)

faire

inst₁;

inst₂;

⋮

inst_n;

finpour;

Syntaxe en Matlab

```
for i=1:10
```

```
inst1;
```

```
.
```

```
.
```

```
instn;
```

```
end
```

Boucle pour II

Somme des entiers

$$S = 1 + 2 + \dots + N$$

eg. $N = 3$

$$S = 1 + 2 + 3 = 6$$

ou $N = 5$

$$S = 1 + 2 + 3 + 4 + 5 = 15$$

Solution

$$S = \sum_{i=1}^N i,$$

```
N=input('N = ');  
s=0;  
for i=1:N  
s=s+i;  
end  
s
```

Boucle pour II

Somme des entiers

$$S = 1 + 2 + \dots + N$$

eg. $N = 3$

$$S = 1 + 2 + 3 = 6$$

ou $N = 5$

$$S = 1 + 2 + 3 + 4 + 5 = 15$$

Solution

$$S = \sum_{i=1}^N i,$$

```
N=input('N = ');  
s=0;  
for i=1:N  
s=s+i;  
end  
s
```

Boucle pour II

Somme des entiers

$$S = 1 + 2 + \dots + N$$

eg. $N = 3$

$$S = 1 + 2 + 3 = 6$$

ou $N = 5$

$$S = 1 + 2 + 3 + 4 + 5 = 15$$

Solution

$$S = \sum_{i=1}^N i,$$

```
N=input('N = ');  
s=0;  
for i=1:N  
s=s+i;  
end  
s
```

Boucle Tant que

Syntaxe en Algorithmique

```
tant que conditions faire  
    insts;  
fin tq
```

Syntaxe en Matlab

```
while conditions  
    insts;  
end
```

Boucle Tant que II

Somme des entiers

$$S = 1 + 2 + \dots + N$$

eg. $N = 3$

$$S = 1 + 2 + 3 = 6$$

ou $N = 5$

$$S = 1 + 2 + 3 + 4 + 5 = 15$$

Solution

$$S = \sum_{i=1}^N i,$$

```
i=1;  
s=0;  
while i < N  
s=s+i;  
i=i+1;  
end
```

Boucle Tant que II

Somme des entiers

$$S = 1 + 2 + \dots + N$$

eg. $N = 3$

$$S = 1 + 2 + 3 = 6$$

ou $N = 5$

$$S = 1 + 2 + 3 + 4 + 5 = 15$$

Solution

$$S = \sum_{i=1}^N i,$$

```
i=1;  
s=0;  
while i < N  
s=s+i;  
i=i+1;  
end
```

Boucle Tant que II

Somme des entiers

$$S = 1 + 2 + \dots + N$$

eg. $N = 3$

$$S = 1 + 2 + 3 = 6$$

ou $N = 5$

$$S = 1 + 2 + 3 + 4 + 5 = 15$$

Solution

$$S = \sum_{i=1}^N i,$$

```
i=1;  
s=0;  
while i < N  
s=s+i;  
i=i+1;  
end
```

Plan

Introduction

Structures de contrôle

Structures itératives

Conclusion