

Université Abderrahmane MIRA - Bejaia

Faculté de Technologie

Département Technologie



Notion d'algorithme et de programme

Dr. A. OUARET



ahmed.ouaret@univ-bejaia.dz

Laboratoire de Technologie Industrielle et de l'Information,
Faculté de Technologie
Université de Bejaia, 06000 Bejaia, Algérie



II.4. Représentation des Algorithmes

Un algorithme peut être représenté selon plusieurs notations. Les deux plus couramment utilisées dans l'enseignement de l'informatique sont :

Organigramme

Une représentation **graphique** utilisant des formes standardisées :



Ovale
début / fin



Rectangle
traitement



Losange
décision



Parallélogramme
entrée / sortie

Pseudocode

Une description **structurée** en français (ou en anglais) utilisant des mots-clés tels que :

SI

TANTQUE

POUR

DÉBUT

FIN

DÉBUT

// exemple

SI note $\geq$ 10 **ALORS**

Écrire("Admis")

FINSI

FIN

Représentation par organigramme

- 1 Un organigramme est une **représentation graphique** d'une solution, équivalente à un algorithme : chaque action a son **symbole standardisé**.
- 2 On lui préfère généralement le **pseudocode**, surtout si le problème est complexe ou comporte des structures imbriquées.

Organigramme

Les principaux **inconvénients** des organigrammes sont :

- 1 Lorsque l'organigramme est long et s'étend sur plusieurs pages, la lisibilité diminue considérablement.
- 2 Les flèches représentant le flux d'exécution peuvent se croiser et se chevaucher, rendant le diagramme confus et difficile à suivre.
- 3 Les organigrammes sont généralement plus difficiles à lire, à rédiger et à maintenir qu'un algorithme bien structuré.

Symboles des Organigrammes

Les symboles utilisés dans les organigrammes sont illustrés et décrits dans le **Tableau II.1** ci-dessous.

Symbole	Nom	Description
 Ovale / Ellipse	Terminal	Marque le Début et la Fin de l'organigramme. Tout organigramme doit avoir exactement un Début et une Fin.
 Parallélogramme	Entrée / Sortie	Représente les opérations d' entrée (Lire — recevoir des données de l'utilisateur) et les opérations de sortie (Écrire — afficher les résultats).
 Rectangle	Traitement	Représente les calculs , les affectations et toutes les étapes de traitement des données (ex. $x \leftarrow a + b$).
 Losange	Décision	Représente les tests et les décisions . La condition est écrite à l'intérieur du losange. Il possède exactement deux sorties : une pour VRAI et une pour FAUX.
 Flèche	Ligne de flux	Indique l' ordre d'exécution et relie les symboles pour montrer la direction du flux logique.

Exemple

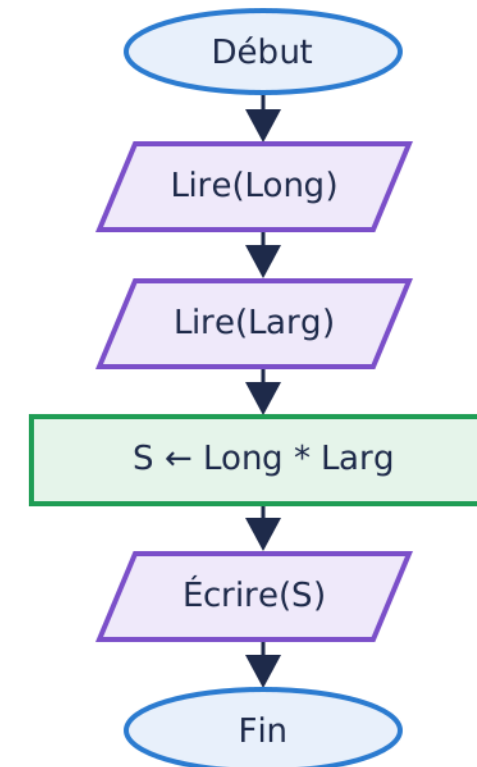
Surface d'un rectangle

Lire la **longueur** et la **largeur** d'un rectangle, puis calculer et afficher sa **surface**.

Algorithme

```
Algorithme SurfaceRectangle ;  
Variable Long, Larg, S : Réel ;  
Début  
  Lire(Long) ;  
  Lire(Larg) ;  
  S ← Long * Larg ;  
  Écrire(S) ;  
Fin.
```

Organigramme



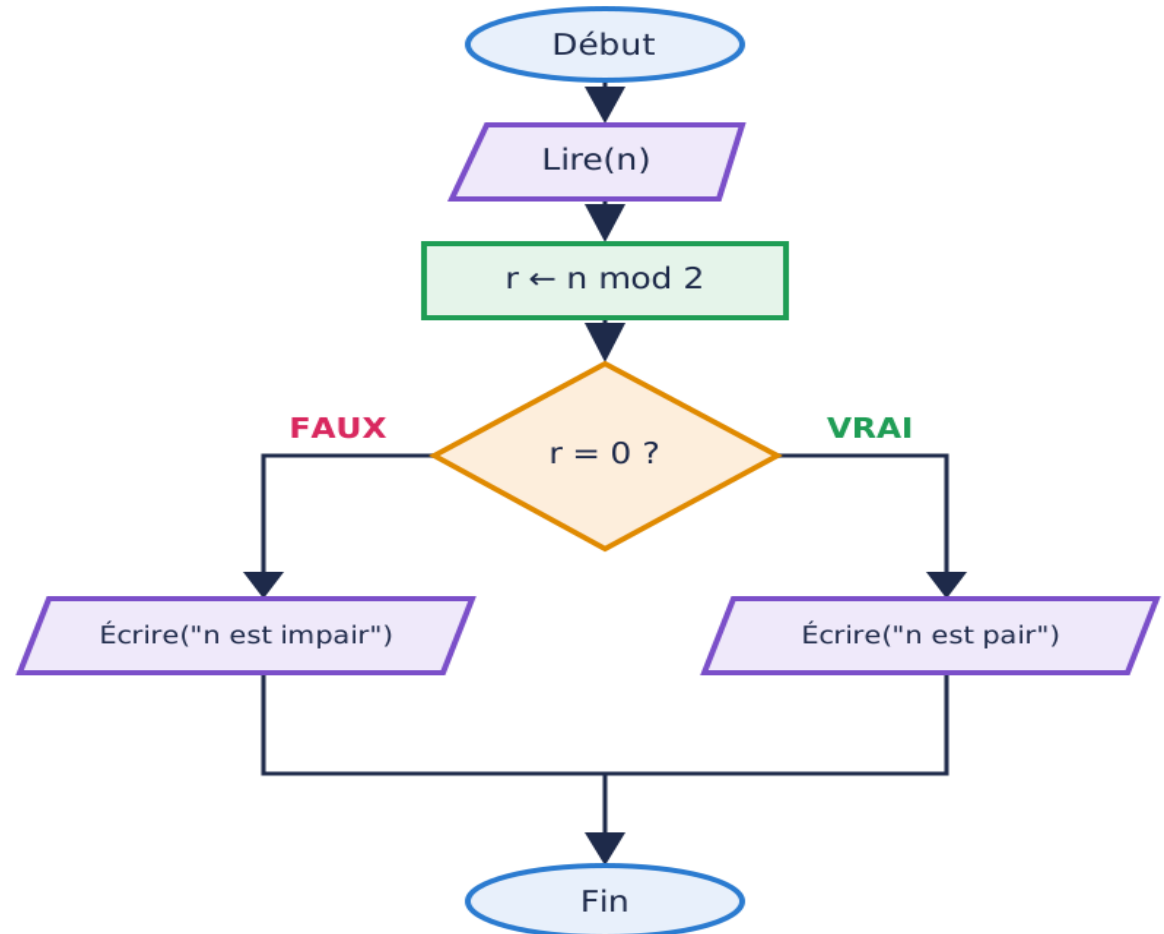
Nombre pair ou impair ?

Lire un entier **n**, puis afficher s'il est **pair** ou **impair**.

Algorithme

```
Algorithme ParitéNombre ;  
Variable n, r : Entier ;  
Début  
  Lire(n) ;  
   $r \leftarrow n \bmod 2$  ;  
  Si (r = 0) Alors  
    Écrire("n est pair") ;  
  Sinon  
    Écrire("n est impair") ;  
  FinSi  
Fin.
```

Organigramme



Travail à faire

De l'algorithme à l'organigramme

Soit l'algorithme suivant. **Dessiner l'organigramme correspondant** en utilisant les symboles standards.

Algorithme

```
Algorithme Cercle ;  
Constante PI = 3.14 ;  
Variable R, P, S : Réel ;  
Début  
    Lire(R) ;  
    P ← 2 * PI * R ;  
    S ← PI * R * R ;  
    Écrire(P) ;  
    Écrire(S) ;  
Fin.
```

Symboles à utiliser (Tableau II.1)



Ovale : Début / Fin



Parallélogramme : Lire / Écrire



Rectangle : calculs, affectations



Flèche : ordre d'exécution



Dessiner l'organigramme ici

Solution

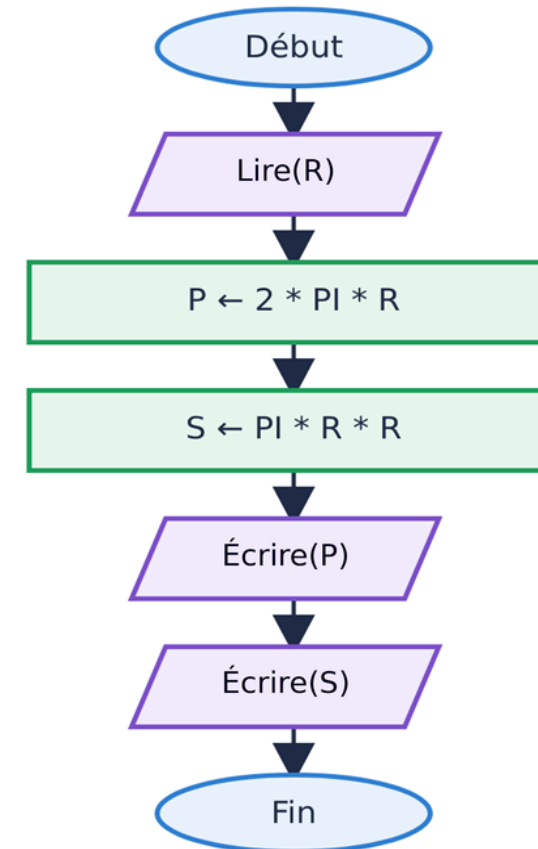
Corrigé : organigramme de l'algorithme Cercle

Lire le **rayon R**, puis calculer et afficher le **périmètre P** et la **surface S**.

Algorithme

```
Algorithme Cercle ;  
Constante PI = 3.14 ;  
Variable R, P, S : Réel ;  
Début  
    Lire(R) ;  
    P ← 2 * PI * R ;  
    S ← PI * R * R ;  
    Écrire(P) ;  
    Écrire(S) ;  
Fin.
```

Organigramme



Exercice

Énoncé → Algorithme → Programme C

Énoncé du problème → Algorithme → Programme en langage C

Écrire un **algorithme**, puis traduisez-le en **programme C**, pour chacun des problèmes suivants :

1 **Permuter** entre les deux variables X et Y ?

2 **Permuter** entre les trois variables X, Y et Z de telle sorte que la valeur de X soit dans Y, celle de Y dans Z et la valeur de Z dans X ?

3 Calculer le **quotient** et le **reste** de la division euclidienne de **A** par **B** (A et $B > 0$) ?

4 Calculer la **somme** de a et b et le **produit** de b et c ?

5 Calculer la **valeur absolue** et le **carré** d'un nombre réel ?

6 Lire les notes de trois matières (**N1** , **N2** et **N3**) ensuite calculer et afficher leur **moyenne** **M** ?
Modifier l'algorithme dans le cas où des **coefficients** (**C1** , **C2** et **C3**) sont attribués aux trois matières.

Exercice N° 03

Énergie cinétique

Écrire un **algorithme**, puis sa traduction en **langage C**, qui lit la **masse** d'un objet en kilogrammes et sa **vitesse** en m/s, puis calcule et affiche son **énergie cinétique** en joules.

NB : L'énergie cinétique est calculée par la formule suivante :

$$Ec = \frac{1}{2} mV^2 ;$$

m : masse de l'objet en kg ;

v : vitesse de l'objet en m/s ;

Ec : énergie cinétique en joules (J).

Données :

Masse de l'objet : **m = 4 kg**

Vitesse de l'objet : **V = 3 m/s**

A hand in a dark suit jacket and white striped shirt cuff points its index finger towards a central white card with the word 'Merci' written on it. The card is surrounded by several other similar white cards, some of which are slightly offset or overlapping, creating a sense of depth. The background is a solid dark blue.

Merci