

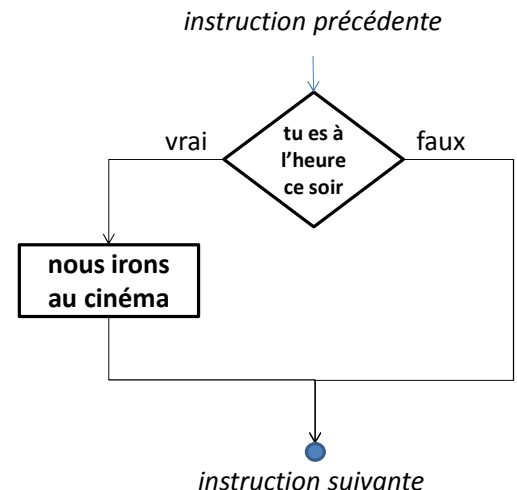
Notion d'exécution conditionnelle ou avec alternative

Dans une procédure, il arrive que certains groupes d'opérations ne soient pas exécutés systématiquement. Leur exécution dépend d'une condition.

Dans la vie courante, on est très souvent confronté à cette situation :

Par exemple :

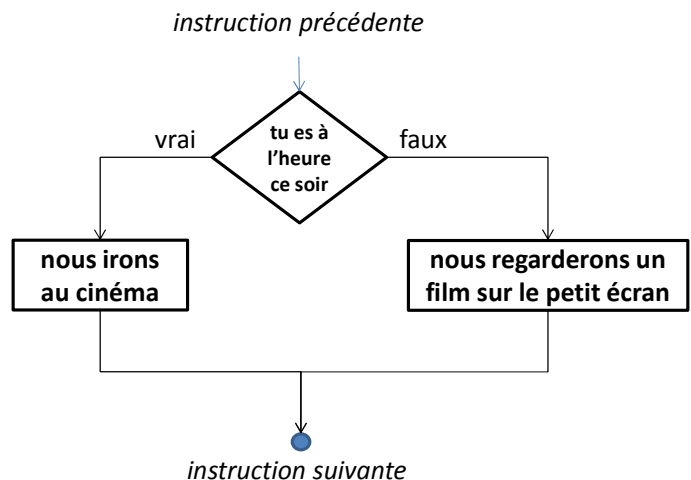
- **SI** tu es à l'heure ce soir **ALORS** nous irons au cinéma.
 - o la condition, exprimée sous forme d'une expression logique, est : «tu es à l'heure ce soir»
 - o si cette condition est vraie, alors «nous irons au cinéma»
 - o dans le cas contraire, on ne fait rien
- **SI** ta moyenne est supérieure à 20, **ALORS** tu auras ta photo dans le journal local
 - o la condition, exprimée sous forme d'une expression logique, est : «ta moyenne est supérieure à 20»
 - o si cette condition est vraie, alors «tu auras ta photo dans le journal local»
 - o dans le cas contraire, rien ne se passe



Il arrive aussi qu'on ait une alternative dans le cas où la condition n'est pas vérifiée :

Par exemple :

- **SI** tu es à l'heure ce soir **ALORS** nous irons au cinéma **SINON** nous regarderons un film sur le petit écran
 - o la condition, exprimée sous forme d'une expression logique, est : «tu es à l'heure ce soir»
 - o si cette condition est vraie, alors «nous irons au cinéma»
 - o dans le cas contraire, «nous regarderons un film sur le petit écran»
- **SI** mes économies me le permettent **ALORS** nous partirons sur la côte d'Azur **SINON** nous profiterons de la belle plage de Calais
 - o la condition, exprimée sous forme d'une expression logique, est : «mes économies me le permettent»
 - o si cette condition est vraie, alors «nous partirons à Nice»
 - o dans le cas contraire, «nous profiterons de la belle plage de Calais»



Structure conditionnelle

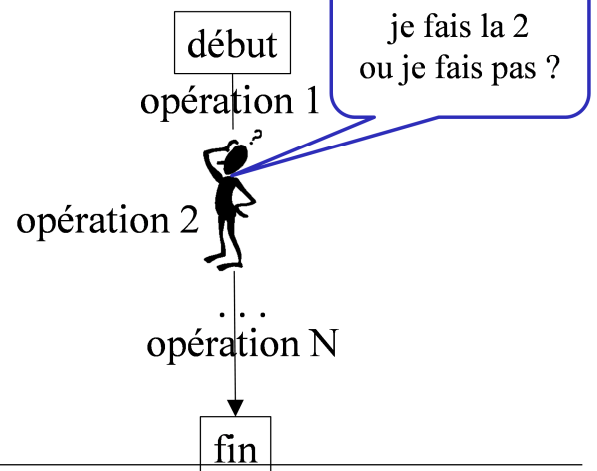
Choix d'exécuter ou pas un ensemble d'opérations en fonction d'une condition

SI (condition)

ALORS

les opérations à réaliser
quand la condition est vraie

finSi



Conditions

Comparaison entre 2 valeurs en utilisant les opérateurs de comparaison : le résultat de la comparaison est une valeur booléenne : Vrai ou Faux

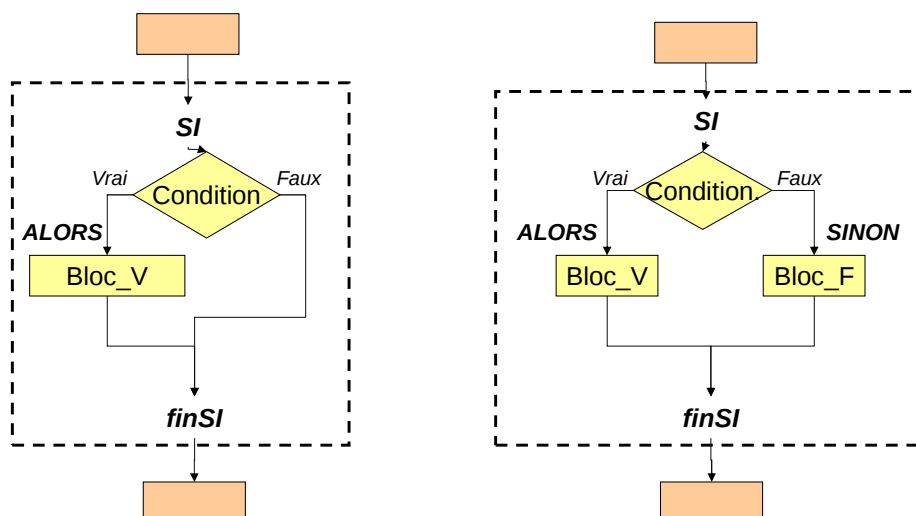
Exemple : mt est un nombre (entier ou réel), reponse est une chaine :

=	<>	<	<=	>	>=
égalité	différence	inférieur	inférieur ou égal	supérieur	supérieur ou égal
(mt = 0)	(reponse <> "oui")	(mt < 100)	(mt <= 100)	(mt > 0)	(mt >= 0)

Pour les comparaisons plus complexes, utilisation de connecteurs ET et OU : le résultat de la comparaison est une valeur booléenne : Vrai ou Faux

Exemple : mt est un nombre (entier ou réel), reponse est une chaine :

((mt = 0) ET (reponse <> "oui"))	((mt = 0) OU (reponse <> "oui"))
Vrai si mt est égal 0 ET si reponse est différent de "oui", Faux sinon	Vrai si mt est égal 0 OU bien si reponse est différent de "oui", Faux sinon



Exemple : calculer la moyenne de 3 notes demandées à l'utilisateur et donner une appréciation sur sa moyenne

On souhaite calculer la moyenne de 3 notes demandées à l'utilisateur

DECLARATIONS

CONSTANTES

VARIABLES

```
note1 : réel ← 0 // 1ere note, saisi
note2 : réel ← 0 // 2eme note, saisi
note3 : réel ← 0 // 3eme note, saisi
som ← : réel 0 // somme des notes, calculé
moy ← : réel 0 // moyenne des notes, calculé
```

DEBUT (des opérations)

// donner une valeur aux notes

AFFICHER « donner 3 notes : »

SAISIR note1 , note2 , note3

// Calculer la somme

som ← note1 + note2 + note3

moy ← som / 3

// donner le résultat

AFFICHER « la moyenne est », moy

// donner une appréciation

SI (moyenne < 10)

ALORS

AFFICHER « c'est insuffisant ! »

AFFICHER « il faut fournir d'avantage de travail ! »

finSI

FIN (des opérations)

Exécution conditionnelle

Exemple : Calculer un montant à payer en fonction d'une quantité commandée

On souhaite avoir une procédure qui nous donne un montant à payer à partir d'un nombre de produits achetés et d'un prix unitaire. On sait qu'à partir de 100 unités, un rabais de 10% est appliqué sur le montant.

DECLARATIONS

CONSTANTES

TAUX_RABAIS : réel $\leftarrow$ 0.1 // taux de rabais accorde
QTE_RABAIS : réel $\leftarrow$ 100 // quantité minimale pour le rabais

VARIABLES

nbre : entier $\leftarrow$ 0 // nombre de produits achetés, saisi
prix : réel $\leftarrow$ 0 // prix unitaire du produit, saisi
montant : réel $\leftarrow$ 0 // montant initiale, calculé
rabais : réel $\leftarrow$ 0 // montant du rabais calculé,
aPayer : réel $\leftarrow$ 0 // montant à payer, calculé, résultats

DEBUT (des opérations)

```
// demander la saisie de nbre et prix
AFFICHER « donnez, dans l'ordre, le nombre et le prix unitaire : »
SAISIR nbre, prix

// Calculer montant
montant  $\leftarrow$  nbre * prix

// si la quantité est suffisante, déterminer le rabais
rabais  $\leftarrow$  0
SI (nbre  $\geq$  QTE_RABAIS)
  ALORS
    rabais  $\leftarrow$  montant * TAUX_RABAIS
  finSI

// calculer le montant à payer
aPayer  $\leftarrow$  montant - rabais

// afficher le rabais si il est différent de 0
AFFICHER « A payer », aPayer
SI (rabais  $\neq$  0)
  ALORS
    AFFICHER « déduction faite du rabais de », rabais
  finSI
```

FIN (des opérations)

Structure conditionnelle avec alternative

Choix d'exécuter un ensemble d'opérations en fonction si une condition est vraie ou un autre ensemble d'opérations dans le cas contraire.

SI (condition)

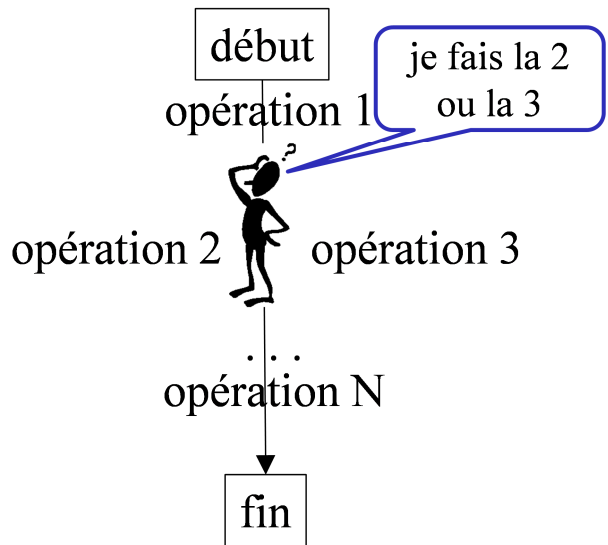
ALORS

les opérations à réaliser
quand la condition est vraie

SINON

les opérations à réaliser
quand la condition est fausse

finSi



Rappel Excel : SI(condition; opérations quand la condition est vraie; opérations sinon)

Exemple : calculer la moyenne de 3 notes demandées à l'utilisateur et donner une appréciation sur sa moyenne

On souhaite calculer la moyenne de 3 notes demandées à l'utilisateur

```

DECLARATIONS
  CONSTANTES
    decimal TAUX_TAXE ← 0.206
  VARIABLES
    note1 : réel ← 0 // 1ere note, saisi
    note2 : réel ← 0 // 2eme note, saisi
    note3 : réel ← 0 // 3eme note, saisi
    som : réel ← 0 // somme des notes, calculé
    moy : réel ← 0 // moyenne des notes, calculé
DEBUT (des opérations)
  // donner une valeur aux notes
  AFFICHER « donner 3 notes : »
  SAISIR note1 , note2 , note3
  // Calculer la somme
  som ← note1 + note2 + note3
  moy ← som / 3
  // donner le résultat
  AFFICHER « la moyenne est », moy
  // donner une appréciation
  SI (moyenne < 10)
    ALORS
      AFFICHER « c'est insuffisant ! »
    SINON
      AFFICHER « c'est bon, ça passe ! »
  finSI
FIN (des opérations)
  
```

Exemple : Déterminer le lieu de vacances en fonction des économies

On souhaite avoir une procédure qui nous permettra de décider du lieu de vacances en fonction du montant des économies : à partir de 1000 euros, on part sur la côte d'azur à Nice, sinon ce sera à Calais (belle plage aussi...). On affichera la ville de destination comme résultat.

```
DECLARATIONS
  CONSTANTES
    MINI_ECO : réel ← 1000 // minimum d'économies
    VILLE1 : chaîne ← « Nice » //
    VILLE2 : chaîne ← « Calais » //
  VARIABLES
    eco : réel ← 0 // montant des économies, saisi
    destination : chaîne ← « » // ville de destination, calculé
DEBUT (des opérations)
  // demander la saisie de eco

  // Déterminer la destination

  // donner le résultat
FIN (des opérations)
```

Imbrication des décisions

Les décisions peuvent être plus complexes et chaque choix peut nécessiter une nouvelle décision : on obtient alors des structures imbriquées les unes dans les autres

Par exemple ici, pour déterminer l'appréciation en fonction de la moyenne

```
...  
SI (moyenne < 10)  
  ALORS  
    AFFICHER « c'est insuffisant ! »  
  SINON  
    SI (moyenne < 12)  
      ALORS  
        AFFICHER « en progrès ! »  
      SINON  
        SI (moyenne < 14)  
          ALORS  
            AFFICHER « pas mal ! »  
          SINON  
            AFFICHER « super ! »  
          finSI  
        finSI  
      finSI  
    finSI  
  finSI  
...
```

Il est alors indispensable de bien mettre en évidence cette imbrication par :

- l'utilisation de l'indentation à chaque nouvelle décision
- l'utilisation d'une barre verticale qui met en évidence le début et la fin de chaque décision

Exemple : Programme de fidélité d'une compagnie aérienne

Des points de fidélité sont accordés et sont transformables en billets d'avion gratuits. Chaque réservation concerne un aller simple (1 trajet) ou un aller-retour (2 trajets). Le compte de points est nominatif et le décompte des points est établi par passager.

Conditions d'attribution des points :

- lors de l'achat d'un billet, 15 pts sont obtenus dès que le montant de l'achat est supérieur ou égal à 50 euros; sinon c'est 5 points qui sont attribués.
- si l'enregistrement est effectué en ligne le nombre de points est doublé par trajet.
- en utilisant la carte bleue de la compagnie, 10 points supplémentaires sont accordés.

Ecrire l'algorithme permettant de déterminer le nombre de points accordés suite à une réservation, et d'afficher ce nombre de points.

Vous utiliserez les noms de variables suivants : `montantRésa`, `nbTrajets`, `enregsLigne`, `CBCie` et `nbPTS`.

Structure de choix multiple

Dans une procédure on écrira ces tests sous la forme suivante :

SELON variable

CAS valeur1, valeur2

les opérations à réaliser

CAS > valeur2 :

les opérations à réaliser

...etc.

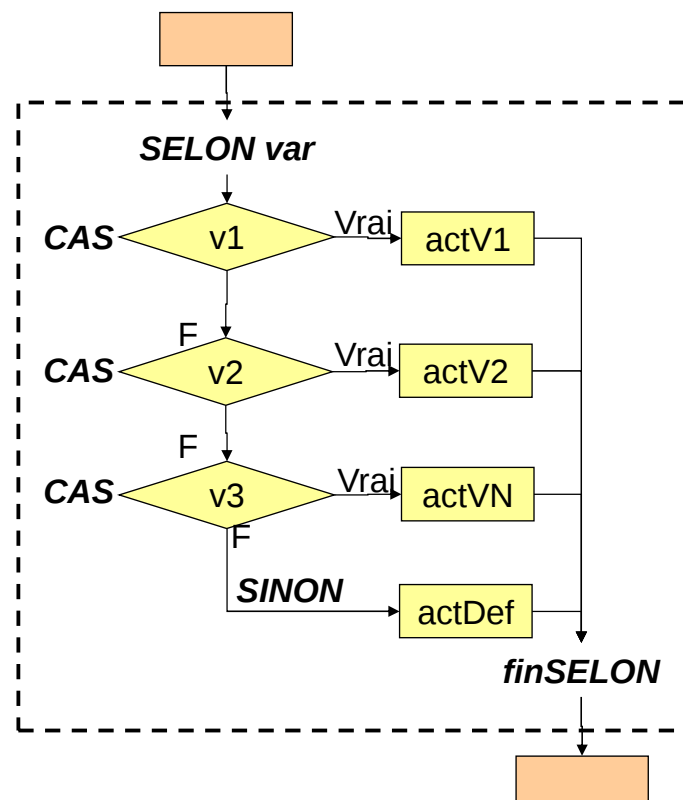
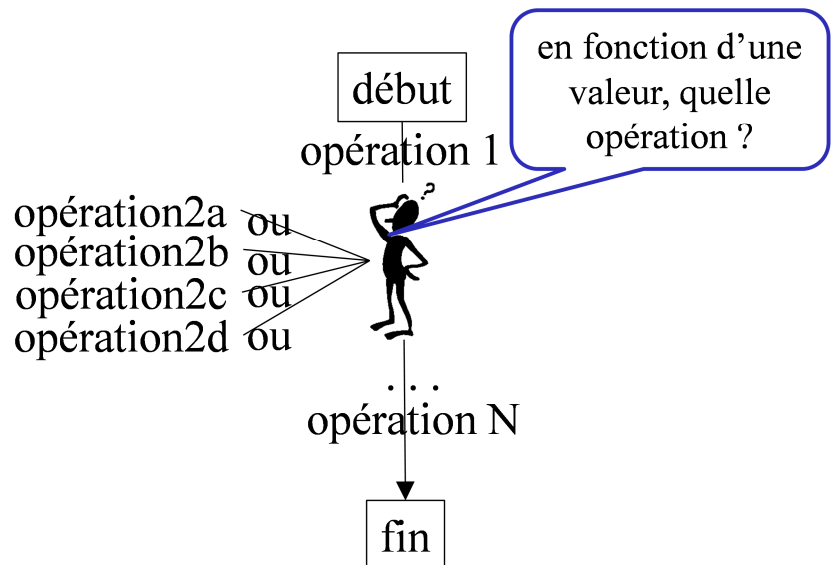
AUTRES CAS :

les opérations à réaliser

quand aucun cas

finSELON

En fonction de la valeur de la variable, **un seul cas sera réalisé.**



Exemple : demander une moyenne et donner la mention

DECLARATIONS

```

VARIABLES
    moyenne : réel ← 0 // moyenne des notes, saisi
    mention : chaîne ← « » // mention, calcul
DEBUT (des opérations)
    demander la saisie de moy
    AFFICHER « donner la moyenne : »
    SAISIR moyenne
    donner une appréciation
    SELON moyenne
        CAS < 12
            mention ← « »
        CAS < 14
            mention ← « assez bien »
        CAS < 16
            mention ← « bien »
        AUTRES CAS
            mention ← « très bien »
    finSELON
    AFFICHER "la mention est ", mention
FIN (des opérations)
    
```

Exemple : déterminer le taux d'amortissement dégressif

```

DECLARATIONS
    CONSTANTES
        COEF1 : réel ← 1.25
        COEF2 : réel ← 1.75
        COEF3 : réel ← 2.25
    VARIABLES
        duree : entier ← 0 // duree, saisi
        coef : réel ← 0 // coefficient, calcul
        taux : réel ← 0 // taux, calcul
DEBUT (des opérations)
    // demander la saisie de la durée
    AFFICHER « donner la duree : »
    SAISIR duree
    // déterminer le coefficient selon la duree
    SELON duree
        CAS 3,4
            coef ← COEF1
        CAS 5,6
            coef ← COEF2
        CAS > 6
            coef ← COEF3
        AUTRES CAS
            coef ← 0
    finSELON
    // calculer puis afficher le taux
    taux ← (100 / duree) * coef
    AFFICHER « le taux est de », taux
FIN (des opérations)
    
```

1.25 pour 3 et 4 ans
1.75 pour 5 et 6 ans
2.25 au-delà
sinon 0