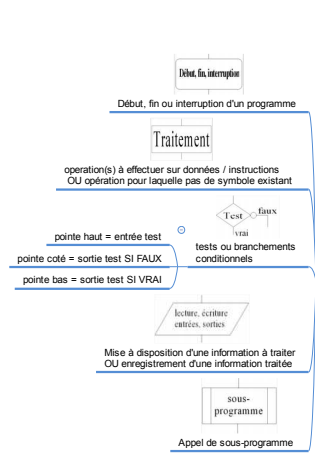
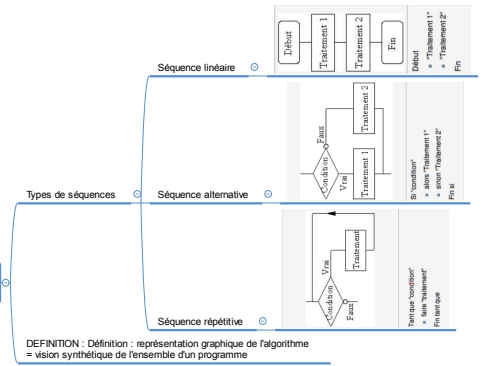


NFA031 - Algorithmique, pseudo-code

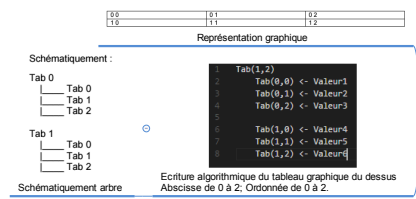
Algorigrammes (graphisme)



Algorigrammes (séquences)



Tableaux multidimensionnels



Fonctions et procédures (suite)

fonction RepOùNon(question en chaîne de caractères) en chaîne de caractères

```
fonction RepOùNon(question en chaîne de caractères) en chaîne de caractères
Début
  TantQue Réponse <> "Oui" et Réponse <> "Non"
    Écrire question
    Lire Réponse
  FinTantQue
  Renvoyer Réponse
Fin
```

Nous pouvons également utiliser le passage de paramètre (ou argument)...

Nous déclarons ici une fonction qui attend un paramètre de type chaîne de caractères.

```
... Rep3 = RepOùNon("Êtes-vous marié ?")
... Rep1 = RepOùNon("Avez-vous des enfants ?")
...
```

Notre procédure principale peut devenir

```
fonction MaFonction()
  Var World en chaîne de caractères
  World <- "Hello world"
  Renvoyer World
Fin
```

Procédure MaProcédure

```
Procédure MaProcédure
  Début
    Écrire "Hello World"
  Fin
```

DEFINITION : Différences

- Une procédure ne renvoie pas de résultat alors qu'une fonction renvoie obligatoirement un résultat.
- Une procédure peut calculer plusieurs résultats, une fonction ne peut renvoyer qu'un seul résultat.

PROCEDURE

Fonctions

L'intérêt est de ne pas réécrire x fois le même code mais d'exploiter celui existant une seule fois.

OBJECTIF

DEFINITION ensemble d'instructions prédéfinies.

REGLÉ ECRITURE

Nom_de_ma_fonction(paramètre1,paramètre2,...)

Une fonction comporte un nom puis entre parenthèses les paramètres attendus, s'il n'y en a aucun les parenthèses restent obligatoires.

Renvoi : 4

Ent(Numérique) renvoi la partie entière d'un nombre

Alea() renvoi un nombre aléatoire compris entre 0 et 1

Renvoi : 6

Len("Durand")

Len(chaine) : renvoie le nombre de caractères d'une chaîne.

Len attend donc un argument sous la forme d'une chaîne de caractères.

Mid(chaine,n1,n2) : renvoie un extrait de la chaîne, commençant au caractère n1 et faisant n2 caractères de long.

Left(chaine,n) : renvoie les n caractères les plus à gauche dans chaîne.

Right(chaine,n) : renvoie les n caractères les plus à droite dans chaîne.

Trouve(chaine1,chaine2) : renvoie un nombre correspondant à la position de chaîne2 dans chaîne1. Si chaîne2 n'est pas comprise dans chaîne1, la fonction renvoie zéro.

Algorithme à transformer en fonction

```
Var Rep1, Rep2, Rep3 en chaîne de caractères
Début
  Écrire "Êtes-vous marié ?"
  Lire Rep1
  TantQue Rep1 <> "Oui" et Rep1 <> "Non"
    Écrire "Répétez Oui ou Non"
  FinTantQue
  Lire Rep1
  Écrire "Avez-vous des enfants ?"
  Lire Rep2
  TantQue Rep2 <> "Oui" et Rep2 <> "Non"
    Écrire "Répétez Oui ou Non"
  FinTantQue
  Lire Rep2
  Écrire "Avez-vous des enfants ?"
  Lire Rep3
  Fin
```

Structure et tableaux structurés

Nom 1, Prénom 1, Tel 1, Mail 1

Nom 2, Prénom 2, Tel 2, Etc.

Exemple réalisation d'un annuaire

```
Structure Annuaire
  Nom en chaîne de caractères
  Prenom en chaîne de caractères
  Tel en chaîne de caractères
  Mail en chaîne de caractères
Fin Structure
```

1. Déclaration de la variable structurée Annuaire

```
Var Personne en Annuaire
```

2. Déclaration de la variable correspondant à la structure (de type enregistrement)

```
Personne <- "Durand", "Pierre", "0442012345", "pierre.durand@truc.com"
```

3. Assignment de valeurs à une personne

```
Personne.Nom <- "Durand"
Personne.Prenom <- "Pierre"
Personne.Tel <- "0442012345"
Personne.Mail <- "pierre.durand@truc.com"
```

3. Autre méthode d'assignment de valeurs

```
Tab_annuaire(N).Nom
Tab_annuaire(N).Prenom
Tab_annuaire(N).Tel
Tab_annuaire(N).Mail
```

Un tableau de données structurées (tableau d'enregistrements) sera donc de la forme (pour une série de N enregistrement)

Dans cet algorithme on observe la répétition de l'analyse de la réponse, on demande à chaque fois de répondre par oui ou par non et on contrôle cette réponse. Nous pouvons donc isoler ce contrôle au sein d'une fonction et faire appel à celle-ci chaque fois que nécessaire.

Algorithme à transformer en fonction

```
fonction RepOùNon() en chaîne de caractères
  Début
    TantQue Réponse <> "Oui" et Réponse <> "Non"
      Écrire "Répétez Oui ou Non"
    Lire Réponse
  FinTantQue
  Renvoyer Réponse
Fin
```

Ecriture de la fonction : Il faut noter l'apparition ici d'un nouveau mot-clé : « Renvoyer » qui retourne le résultat du traitement. Attention, les fonctions doivent être écrites en dehors de la procédure principale.

Déclaration de la fonction : Appelons-là RepOùNon

```
... Écrire "Êtes-vous marié ?"
Rep1 <- RepOùNon()
... Écrire "Avez-vous des enfants ?"
Rep2 <- RepOùNon()
...
```

Reprise de l'exemple