

Algorithmique

L1 math-info, semestre 2

Philippe Langlois
langlois@univ-perp.fr
 Bâtiment B, étage 1, à gauche

Introduction générale

Que va-t-on faire ? Comment ?
 Pré-requis ? Acquis en fin de semestre ?
 Contrôle de connaissance ?

29/03/16 18:45

Algo 2. L1 math-info. PhL

2

Ce qui est acquis du semestre 1

- Variable **affectation** valeur : aujourd'hui = (ou \leftarrow) l1
- Variable et valeur **scalaire** : entier, flottant, booléen, caractère
- Entier, flottants : arithmétique, comparaison
- Booléen : algèbre booléenne : NON, ET, OU, XOR
- Affectation, **affichage** et **lecture** pour tous !
- Choisir : choix simple ou multiple
- Répéter : un nombre de fois fixé ou indéterminé a priori
- Tableau-vecteur : des cases numérotées et contigües de valeurs de même type
- Sous-programme : **fonction**, (**procédure**) pour **structurer** : factoriser, regrouper, éviter de réécrire

29/03/16 18:45

Algo 2. L1 math-info. PhL

3

Ce qu'on va apprendre ce semestre

Etre vraiment à l'aise avec les notions précédentes

Stocker

Tableau, enregistrement, liste
 Piles

Récurtivité

Structure récursive : listes
 Algorithmes récursifs : plein !!!

Rechercher

Algorithmes itératifs et algorithmes récursifs

Trier

Algorithmes itératifs et algorithmes récursifs

29/03/16 18:45

Algo 2. L1 math-info. PhL

4

Ce qu'on va apprendre ce semestre, suite.

Je veux résoudre un problème

- Calculer la moyenne des notes en algo au S1
- Recherche les notes maximale et minimale
- Trier ces notes par ordre croissant

Prouver la correction d'un algorithme

- Est-ce que l'algorithme se termine ?
- Est-ce que l'algorithme calcule bien la/une solution du problème ?

Complexité d'un algorithme

- Combien de "temps" ça va prendre ?
- Combien d'espace mémoire est nécessaire ?

29/03/16 18:45

Algo 2. L1 math-info. PHL

5

Organisation de l'enseignement

Cours-TD

- 16 séances de CM et de TD
- 2 groupes de TD : Mme MC ElJai, mm
- 1^{ère} séance de TD : correction de l'examen de décembre

Contrôle des connaissances

- Contrôle continu à **mi-parcours** en TD : **la semaine de la rentrée des vacances d'hiver**
- Contrôle terminal écrit

Utiliser l'ENT

- Diapos du cours et feuilles de TD, références biblio ou URL

Etat d'esprit pour réussir !

- Ne manquer aucune occasion pour poser des questions
- Ne manquer aucune occasion pour lire, chercher, pratiquer

29/03/16 18:45

Algo 2. L1 math-info. PHL

6

Algorithmique avec ou sans programmation ?

Bilan du semestre 1

- Vos impressions, vos suggestions

Programmer ...

- aide à comprendre les détails d'un algorithme et son comportement sur des cas moyens ou extrêmes
- peut faire perdre du temps si on ne maîtrise pas un langage de programmation et son environnement
- ne dispense pas de l'indispensable phase de compréhension :
dérouter à la main, au moins un peu !
- ne prouve rien ! ni correction, ni complexité

Pour le semestre 2

- Qui possède un ordinateur portable ?
- Qui connaît python ? (langage et environnement très faciles d'usage)

29/03/16 18:45

Algo 2. L1 math-info. PHL

7

Notations pour l'algo du S2

Profitions de plusieurs formalismes pour mieux dégager les concepts

29/03/16 18:45

Algo 2. L1 math-info. PHL

8

Notations très très similaires

algo principal, **déclaration**, **affectation**, **comparaison**, boucle **tant que**, test, E/S

```
# Rôle : Afficher les valeurs paires entre 0 et n.
Déclaration
  n : entier # valeur lue au clavier : donnée
  i : entier # compteur de boucle :
intermédiaire
Début
  afficher("Saisir une valeur entière : ")
  lire(n)
  tant que i < n faire
    si i %2=0 alors
      afficher("Oui ", i, " est une valeur
      paire.", retour_a_la_ligne)
    fsi
    i = i+1
  ftantque
Fin
```

```
//rôle : afficher les valeurs paires entre 0 et n
déclare
  n : entier // donnée, lue au clavier
  i : entier // itérateur des valeurs entre 0 et n
début
  afficher("Saisir une valeur entière : ")
  lire(n)
  tant que (i < n) faire
    si (i %2 == 0) alors
      afficher("Oui ", i, " est une valeur
      paire.", retour_a_la_ligne)
    fin si
    i = i+1
  fin tant que
fin
```

Notations très similaires

Fonction, **paramètres**, **zone de déclaration**, tableaux, boucle pour, tests, description-commentaire

```
Fonction Minimum(entrée t : tableau[]
d'entiers,
  entrée n : entier) : entier
#Rôle : identifie le min d'un tableau d'entiers
Déclaration
  i : entier # itérateur de boucle,
intermédiaire
  min : entier # minimum de t, sortie
Début
  si n > 0 alors
    min = t[0]
    pour i de 1 à n-1 faire
      si t[i] < alors
        min = t[i]
      fsi
    fpour
  sinon
    afficher("tableau vide")
  fsi
  retourner(min)
Fin
```

```
fonction Minimum(t[] : in tableau d'entiers,
  n : in entier) retourne entier
//rôle: identifie le min d'un tableau t d'entiers de taille
n
i : entier // indice de parcours de t
min : entier // résultat, minimum de t
début
  si (n > 0) alors
    min = t[0]
    pour i de 1 à n-1 faire // t est indexé de 0 à n-1
      si (t[i] < min) alors
        min = t[i]
      fin si
    fin pour
  sinon
    afficher("tableau vide")
  fin si
  retourner min
fin fonction
```

Autres notations

tableaux et fonctions

Déclaration de tableau :

- la déclaration d'une variable tableau doit fixer :
 - . son identificateur,
 - . sa dimension,
 - . la taille de chacune des dimensions qui est fixée une fois pour toute lors de la déclaration,
 - . les indices de début et de fin (0 par défaut)
 - . et le type des valeurs

Appel de fonction externe :

- Pour utiliser (appeler) une fonction hors de son contexte de définition, il faut la déclarer "comme une variable locale" :
 - . en rappelant son en-tête

```
déclare
//rôle: tableaux et fonction externe
n1, n2 : entiers = 3
matrice[1..n1, 1..n2] : tableau de flottants
cube[2,2,2] : tableau de booléens
nb[8] : tableau d'entiers = [0,0,0,0,0,0,0]
i, j : entiers // itérateurs
fonction F(m : tableau 3D de booléens, n, n2, n3 : entiers)
  retourne entier
  début
    pour i de 1 à n1 faire // ces boucles commencent
      pour j de 1 à n2 faire // à 1 comme matrice
        matrice[i,j] = flottant(i+j) //conversion de type
      fin pour
    fin pour
  fin
  pour i de 0 à 7 faire // commence à 0 comme nb
    nb[i] = F(cube)
  fin pour
fin
```

Récapitulatif des différences de notations

algo principal

- déclare / début / fin
- affectation avec = , comparaison avec ==
- identificateur_de_variable : son_type = valeur_par_défaut // déclaration
- en commentaires : //role (lexique) et description des variables

sous-programme : fonction et procédure

- en-tête : mode des paramètres **in**, **out**, **inout** // entrée, sortie, entrée-sortie
- en-tête : le mode par défaut est in (il peut ne pas être indiqué)
- corps : rappel de l'en-tête pour commencer une "zone declare" : entre fonction(...) et début

Au delà de la syntaxe

Exhiber les blocs logiques de l'algorithme

- blocs logiques: zones déclaratives, boucles, tests, fonctions, E/S
- exhiber : indenter, utiliser des fonctions

Privilégier la lisibilité au dépend de l'écriture

- choisir des identificateurs "qui parlent" ... sans être trop long à écrire non plus
- exemples : n vs. taille, i et j vs. lig et col, f(..) vs. deuxfois(..), ...

Trouver le meilleur des mondes

entre le codage de l'algorithmique
et la syntaxe rigide et parfois hermétique des langages de programmation

29/03/16 18:45

Algo 2. L1 math-info. PhL

13

Synthèse de la séance

Ce qu'il faut retenir !

29/03/16 18:45

Algo 2. L1 math-info. PhL

14

Sur la suite et sa forme

La suite

vision plus globale des notions nouvelles qui seront abordées :

- structures de données composées : tableaux 1D, 2D, enregistrements, listes
- recherche et tri sur ces structures de données
- traitements et structures de données récursives
- preuve de la correction et complexité des algorithmes

Sur la forme

- intérêt d'assister au cours
- se poser des questions ... et nous les poser si besoin !
- QCM 5-10 minutes en début de la prochaine séance
- **Test d'auto-évaluation de l'acquis de la semaine sur l'ENT**

29/03/16 18:45

Algo 2. L1 math-info. PhL

15