



LIFAPR : ALGORITHMIQUE ET PROGRAMMATION RÉCURSIVE

Présentation de l'UE
Modalités de Contrôle des Connaissances

PRÉSENTATION DE L'UE LIFAPR

- Responsables de l'UE & des amphis
 - Marie Lefevre – marie.lefevre@univ-lyon1.fr
 - Nathalie Guin
- Site Web de l'UE
 - <https://perso.liris.cnrs.fr/marie.lefevre/ens/LIFAPR/>
 - Planning (salles, horaires, enseignants)
 - Supports des CMs, sujets et corrigés des TD/TPs
 - Modalités de Contrôle des Connaissances
 - Lien vers ASKER

CONTENU

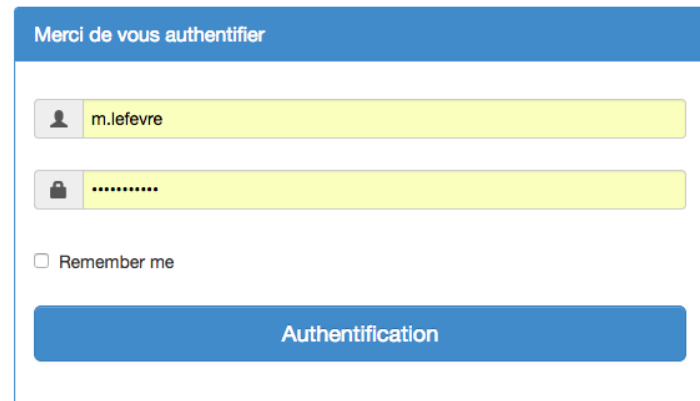
- Notions : récursivité, programmation fonctionnelle
 - structures de données listes et arbres
- Complémentaire à l'UE LIFAPI
 - programmation impérative et itérative
- Utile dans la suite des UE d'algorithmique et programmation (LIFAPSD, LIFAPCD, LIFPF...)
 - afin de choisir entre une approche impérative et une approche récursive pour résoudre un problème
 - pour avoir eu une première présentation des listes chaînées et des arbres

ORGANISATION

- Volume horaire
 - 7,5h de CM (5 séances d'1h30)
 - 22,5h de TD/TP (séances d'1h30 ou 3h)
- Plan « Réussite en Licence »
 - 1 TD/TP de révisions pour préparer le TP noté
 - Des séances de soutien et de tutorat

ASKER : UN OUTIL POUR VOUS ENTRAÎNER

- Exercices simples de révision du cours
- Pour réviser avant le TD/TP et s'auto-évaluer
- Possibilité de refaire plusieurs fois le même exercice avec des données différentes
- <https://asker.univ-lyon1.fr/>
 - Identifiants Lyon 1
 - Tutoriel sur la page web



The screenshot shows a web form for authentication. At the top, a blue header bar contains the text "Merci de vous authentifier". Below this, there are two yellow input fields. The first field has a person icon on the left and contains the text "m.lefevre". The second field has a lock icon on the left and contains a series of dots ".....". Below these fields is a checkbox labeled "Remember me". At the bottom of the form is a large blue button with the text "Authentification".

LISTE DES EXERCICES

LIFAP2 LIFAP1 m.lefevre

 Bases de l'algorithmique

	LIFAP1 - Les structures de contrôle	MARIE LEFEVRE	
	LIFAP1 - La syntaxe en algorithmique	MARIE LEFEVRE	
	LIFAP1 - La boucle POUR pour afficher	MARIE LEFEVRE	
	LIFAP1 - Les étapes de conception	MARIE LEFEVRE	

 Bases du C

**Dans le dossier LIFAR
vous trouverez des dossiers thématiques
contenant chacun plusieurs exercices**

POUR FAIRE UN EXERCICE

 Bases de l'algorithmique

	LIFAP1 - Les structures de contrôle	MARIE LEFEVRE	
	LIFAP1 - La syntaxe en algorithmique	MARIE LEFEVRE	
Demander un nouvel exercice			
	Exercice 1		
	Exercice 2		
	Exercice 3		

LES EXERCICES

LIFAP1 - La syntaxe en algorithmique

Consigne : Répondez aux questions suivantes :

Cochez les types valides en algorithmique :

- ☐ réel
- ☐ caractère
- ☐ booléen
- ☐ bloc
- ☐ multiplication
- ☐ entier

L'exercice


**Valider &
Refaire**

**Plusieurs
questions : valider
chaque question
indépendamment**

LA CORRECTION D'UNE QUESTION

LIFAP2

LIFAP1

 m.lefevre

LIFAP1 - La syntaxe en algorithmique

Consigne : Répondez aux questions suivantes :

Cochez les types valides en algorithmique :

☒ réel

☒ caractère

☐ booléen

☐ bloc

☐ multiplication

☒ entier

Commentaire :

Votre score : 0%

Solution

☒ réel

☒ caractère

☒ booléen

☐ bloc

☐ multiplication

☒ entier

COMPÉTENCES & RECOMMANDATIONS

LIFAP2

ext_test

Exercices

Compétences et recommandations

Mes objectifs

Choix des objectifs

Récupérer les objectifs de l'enseignant

Récupérer mes objectifs

Compétence : connaître les primitives sur les arbres

Intention pédagogique : Revision ②

Mes recommandations

Obtenir de nouvelles recommandations

connaître les primitives sur les arbres

Definition en Schéma d'arbre contenant des symboles

Definition en Schéma d'arbre contenant des nombres

Primitives sur les arbres : leurs noms

Primitives sur les arbres : les types de sortie

Primitives sur les arbres : les types d'entrée

Mon profil

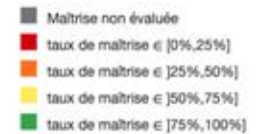
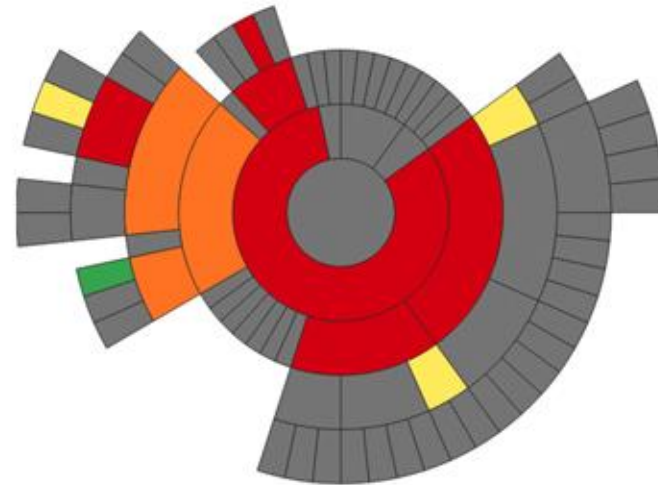
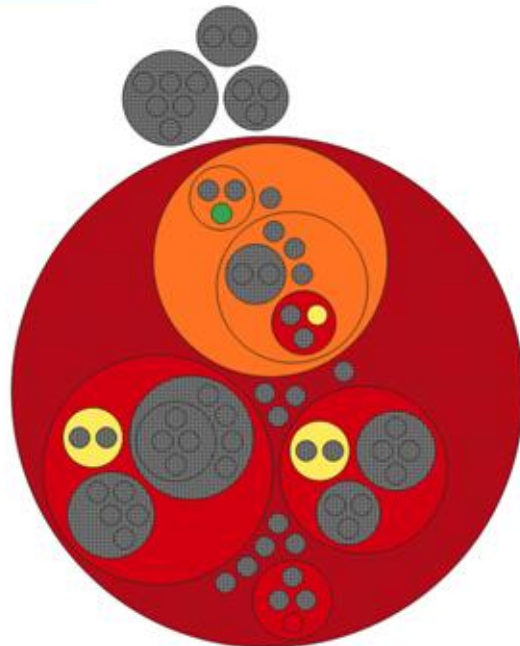
Mettre à jour mon profil

Cacher les ressources

Arborescent

	maîtrise	confiance
○ LIFAP2_2021	-	-
○ Comprendre la récursivité	0%	0%
○ Connaître la définition de la récursivité	0%	0%
○ Reconnaître un algorithme récursif ou non	0%	0%
▶ Savoir écrire une fonction Scheme répondant à une spécification	5%	25%
○ connaître le processus d'évaluation de l'appel à une fonction	0%	0%
▶ connaître les primitives sur les arbres	3%	21%
Exercice : Primitives sur les arbres : leurs descriptions		
Exercice : Primitives sur les arbres : test, accès ou construction ?		
Exercice : Primitives sur les arbres : les types d'entrée		
Exercice : Primitives sur les arbres : les types de sortie		
Exercice : Primitives sur les arbres : leurs noms		
○ connaître les primitives de test sur les arbres	0%	0%
○ connaître la représentation d'un arbre binaire	31%	52%
Exercice : Definition en Schéma d'arbre contenant des symboles		
Exercice : Definition en Schéma d'arbre contenant des nombres		
○ connaître les primitives de construction d'arbres	0%	0%
○ connaître les primitives d'accès aux arbres	0%	0%
○ connaître les structures de conditionnelle alternative	0%	0%
▶ connaître les fonctions prédéfinies	25%	38%
○ connaître les fonctions sur les listes	43%	55%
Exercice : Construction de listes		
Exercice : Primitives sur les listes : type d'entrée		
Exercice : Primitives sur les listes : type de sortie		
Exercice : Primitives sur les listes : description		
○ Savoir choisir les bonnes primitives pour construire une liste	0%	0%
○ connaître la représentation d'une liste	0%	0%
○ connaître les primitives de construction de listes	23%	54%
○ connaître la fonction list	0%	0%
○ connaître la fonction cons	68%	54%
Exercice : Evaluation d'expressions avec la fonction cons		
○ connaître la fonction append	0%	0%
○ connaître les primitives de test sur les listes	0%	0%

PROFIL DE COMPÉTENCES



COMPÉTENCES VISÉES

- C1. Connaître les fonctions d'accès (car, cdr) et de construction (cons, list, append) de listes et savoir les utiliser à bon escient
- C2. Savoir écrire une fonction qui parcourt récursivement une liste pour retourner un booléen, un atome (nombre, symbole), ou une liste
- C3. Reconnaître quand il est nécessaire de mémoriser une valeur et savoir utiliser le let pour le faire
- C4. Savoir écrire une fonction qui retourne une liste de 2 résultats
- C5. Savoir écrire et utiliser un jeu de test
- C6. Savoir reconnaître si une fonction effectue les calculs en remontant ou en descendant. Être capable de compléter une fonction écrite selon l'une ou l'autre des méthodes
- C7. Savoir écrire une fonction qui opère en profondeur sur une liste
- C8. Savoir écrire une fonction qui parcourt récursivement un arbre binaire, pour retourner un booléen, un nombre, une liste ou un arbre

MODALITÉS DE CONTRÔLE DES CONNAISSANCES

- Plusieurs épreuves :
 - CC1 et CC2 sur papier en TD (20% et 30%) C1 C2 C3 C4
 - TP noté en conditions d'examen sur C5 (40%) C1 à C8
- Harmonisation des notes en fin de semestre
- Epreuve de seconde chance : elle remplace une absence en CC ou TP noté, ou la pire des notes