

R1.01.P2 – Initiation au développement

Objectif : algorithmique plus complexe et évaluation de l'efficacité d'un algorithme

&

Types abstraits

Organisation du cours

Organisation de l'enseignement

- 1h30 Cours suivi de (la semaine suivante) :
 - 1 X 1h30 TD en salle banalisée
 - 1h30 TP en salle machine par 1/2 groupe
- TD : préparation du TP mais pas uniquement
- TP : codage en Java d'algorithmes étudiés en TD

Java comme en R1.01.P1, pas d'objet (sauf en fin de ressource)

Poids de l'enseignement dans les compétences

Evaluation de la ressource

- 1 note de TP : poids $1/3$ (attention, TP rendu en retard = malus !)
- 1 note contrôle terminal (en semaine 02) : poids $= 2/3$

SAE S1.02

- 40% de la compétence 2
- « Comparaison d'approches algorithmiques »

Ne rien négliger !!

Poids de l'enseignement dans les compétences

Comp.1
Réaliser un
développement
d'application

Comp.2
Optimiser des
solutions
informatiques

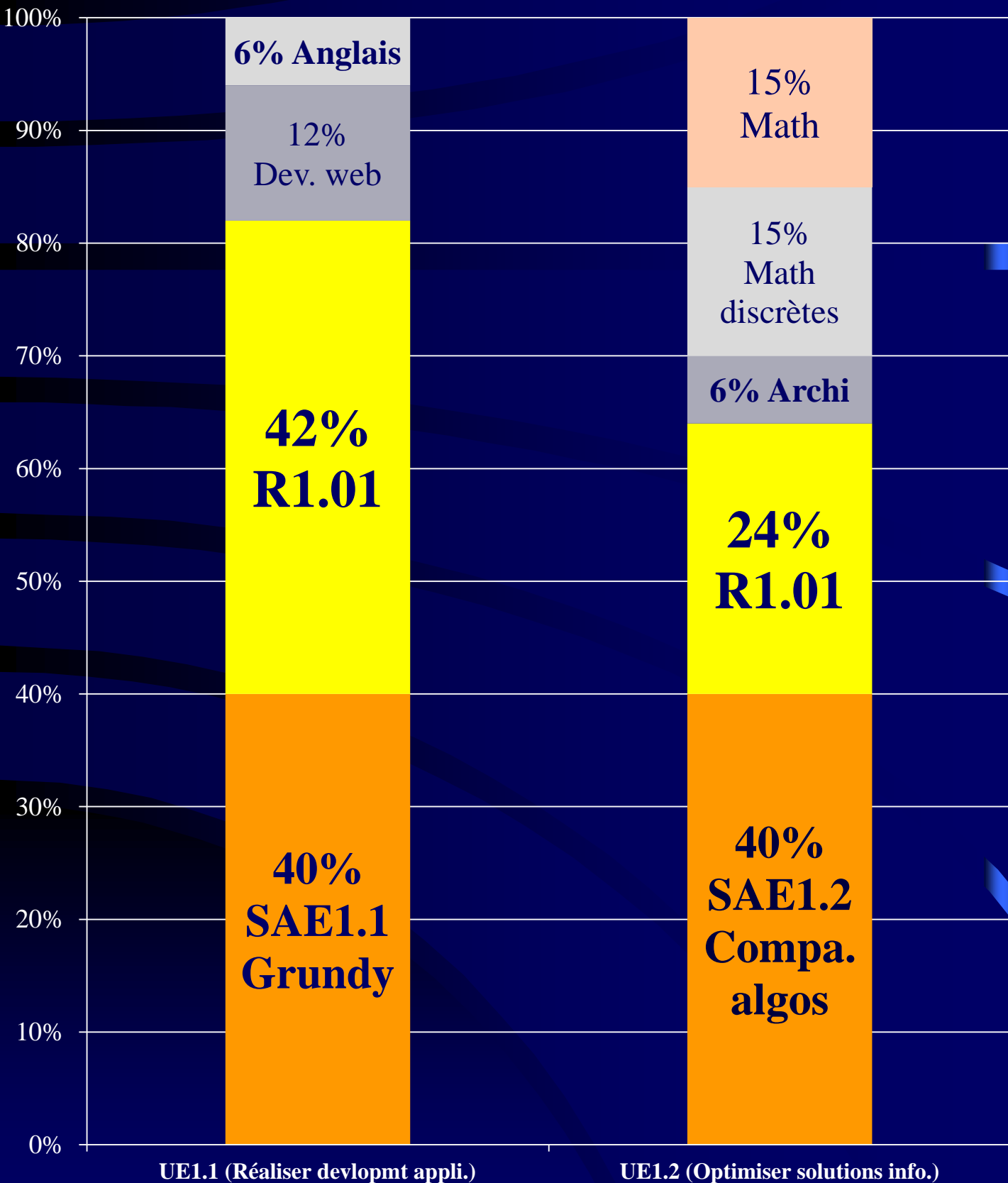
Comp.3
Administrer des
systèmes
informatiques
communicants
complexes

Comp.4
Gérer des
données de
l'information

Comp.5
Conduire un
projet

Comp.6
Se projeter dans
un
environnement
professionnel

Poids de R1.01 et SAE sur les compétences



Organisation de l'enseignement

Organisation sur la période (3 parties) :

- Partie 1 – semaines 45 à 46 : notion de complexité, TP d'algorithmes simples sur des tableaux

Rendu du TP : semaine 46 sur Moodle

- Partie 2 – semaines 47 à 49 : algorithmes de recherche et de tris, étude de la complexité

Rendu du TP : semaine 49 sur Moodle

- Partie 3 – semaines 50 et 01 : Types abstraits et collections

Rendu du TP : semaine 01 sur Moodle

Organisation de l'enseignement

Le cours sur Moodle :

- BUT1/Semestre1/Ressources :
<https://moodle.univ-ubs.fr/course/view.php?id=7518>
- Clé d'inscription obligatoire = R1.01
- JDK11 maximum (JDK8 suffit)
- Organisation semaine par semaine
- Espaces de rendus : dates et heures à respecter impérativement

PLAN du cours

R1.01.P2

1. Tests unitaires de méthodes, récursivité (sem.44)
2. Notion de complexité d'un algorithme (sem.45)
3. Algorithmes de recherche (sem.46)
4. Tri simple et tri rapide (sem.47 & sem.48)
5. Types abstraits (sem.49)
6. Lecture/Ecriture fichier (sem.50)