Mise en place d'une évaluation par compétences pour un cours d'informatique de 3ème année

Nicolas Delestre

INSA
ROUEN NORMANDIE

Contexte

Algorithmique avancée et programmation C

- Cours d'informatique dispensé aux ASI3
- Séance : 21h de CM, 21h de TD et 21h de TP
- Matériel pédagogique : support de cours, QCM d'auto-évaluation, livret d'exercice de TD avec correction, évaluation par les pairs des TP, annales des examens théoriques (avec correction) et pratiques
- Note finale calculée à partir de :
  - **deux examens sur table**
  - un examen machine
  - un projet
- Examens sur table : évaluation descendante

INSA ROUEN NORMANDIE

Objectifs

Évaluation descendante → évaluation par compétences

- Avantages pour l'équipe pédagogique :
  - une élaboration des examens plus réfléchies
  - une évaluation certainement plus objective
  - des décisions de jury plus justifiées
  - une évaluation diagnostique à l'issue du premier examen
- Avantages pour l'étudiant :
  - une meilleure compréhension des attentes des enseignants
  - un meilleur positionnement sur ce qu'il doit (re)travailler
- Méthodologie
  1. définition d'un référentiel de compétences
  2. rédaction et évaluation des examens sur table
  3. production d'un tableau de bord

INSA ROUEN NORMANDIE

Le référentiel de compétences 1 / 2

Rédaction

- Référentiel organisé en catégories, sous-catégories et compétences
- Les catégories sont issues des étapes principales du cycle de développement en V : analyse, conception préliminaire, conception détaillée, développement, tests unitaires
- Chaque compétence a un identifiant
- Les compétences sont pour la plupart des savoir-faire (« savoir... ») et quelques savoirs (« connaître... »)
- Certaines compétences peuvent être évaluées au milieu du semestre
- Le référentiel a évolué une fois au milieu du semestre
- Aujourd'hui il est composé de 70 compétences

INSA ROUEN NORMANDIE

Le référentiel de compétences 2 / 2

Algorithmique avancée et programmation C
Référentiel de compétences

Domaine de compétences	Sous domaine de Compétences	Code	Compétence	À maîtriser pour le partiel
Analyse				
		AN001	Savoir désigner les choses (identifiant significatif)	X
		AN002	Savoir être précis quant aux types de données utilisés	X
		AN003	Connaître le rôle de l'analyse	X
	Analyse descendante			
		AN101	Savoir identifier les entrées et sorties d'un problème	X
		AN102	Savoir décomposer logiquement un problème	X
		AN103	Savoir généraliser un problème	X
		AN104	Savoir si un problème doit être décomposé	X
		AN105	Savoir identifier un problème naturellement récursif (directement ou indirectement)	X
	Type abstrait de données			
		AN201	Savoir identifier les dépendances d'un TAD	X
		AN202	Savoir définir des TAD générique	X
		AN203	Savoir si une opération identifiée fait parti du TAD à spécifier	X
		AN204	Savoir formaliser des opérations d'un TAD	X
		AN205	Savoir formaliser les préconditions d'une opération d'un TAD	X
		AN206	Savoir formaliser des axiomes ou savoir définir la sémantique d'une opération d'un TAD	X
	Collection			
		AN301	Savoir lister les collections usuelles	X
		AN302	Savoir formaliser sous forme de TAD une collection	X
	Graphe			
		AN401	Savoir spécifier les différents types de graphes (étiqueté et/ou valué)	

INSA ROUEN NORMANDIE

Rédaction des examens sur table 1 / 2

Rédaction

- Chaque question de l'examen est composée :
  - du libellé
  - de la liste des compétences évaluées
  - de sa correction
- Le sujet de l'examen contient uniquement le libellé de la question
- La correction de l'examen contient les trois informations
- Un seul fichier source permet de produire ces deux documents (utilisation conjointe de LaTeX, d'un *package* solution, et de l'outil *make*)

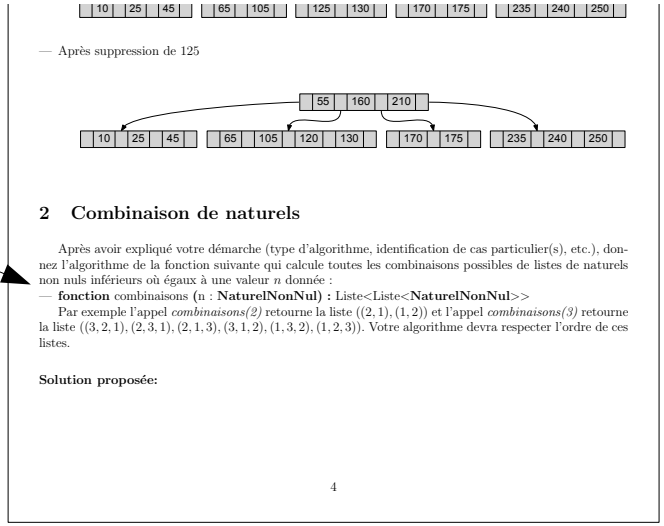
INSA ROUEN NORMANDIE

Rédaction des examens sur table 2 / 2

Sujet / question

Compétences évaluées

Correction



10 | 25 | 45 | 65 | 105 | 125 | 130 | 170 | 175 | 235 | 240 | 250

— Après suppression de 125

55 | 160 | 210

10 | 25 | 45 | 65 | 105 | 120 | 130 | 170 | 175 | 235 | 240 | 250

2 Combinaison de naturels

Après avoir expliqué votre démarche (type d'algorithme, identification de cas particulier(s), etc.), donnez l'algorithme de la fonction suivante qui calcule toutes les combinaisons possibles de listes de naturels non nuls inférieurs où égaux à une valeur n donnée :

— **fonction** combinaisons (n : **NaturelNonNul**) : Liste<Liste<**NaturelNonNul**>>

Par exemple l'appel *combinaisons(2)* retourne la liste $((2,1),(1,2))$ et l'appel *combinaisons(3)* retourne la liste $((3,2,1),(2,3,1),(2,1,3),(3,1,2),(1,3,2),(1,2,3))$. Votre algorithme devra respecter l'ordre de ces listes.

Solution proposée:

4

Compétences évaluées

— Compétences principales :

CD009 Savoir écrire un algorithme qui résout le problème

CD201 Savoir identifier et résoudre le problème des cas non récursifs

CD202 Savoir identifier et résoudre le problème des cas récursifs

— Compétences secondaires :

CD004 Savoir écrire des algos avec le pseudo code utilisé à l'INSA

CD005 Savoir écrire un pseudo code lisible (indentation, identifiant significatif)

CD006 Savoir choisir la bonne itération

Ce problème est naturellement récusif. Le cas d'arrêt est lorsque n vaut 1 et dans ce cas la valeur retournée est $((1))$. Le cas général, pour une valeur n résoud le problème pour la valeur $n-1$, on obtient donc une liste l. Et pour chaque élément de la liste l et pour chaque position possible (de 1 à $longueur(l)+1$), on insère la valeur n.

```
fonction combinaisons (n : NaturelNonNul) : Liste<Liste<NaturelNonNul>>
  Déclaration  i : NaturelNonNul
               listeDeListe, res : Liste<Liste<NaturelNonNul>>
               liste1,liste2 : Liste<NaturelNonNul>
debut
  res ← liste()
  si n=1 alors
    liste1 ← liste()
    inserer(liste1,1,1)
    inserer(res,1,liste1)
  sinon
    listeDeListe ← combinaisons(n-1)
    pour chaque liste1 de listeDeListe
      pour i ←1 à longueur(liste1)+1 faire
        liste2 ← liste1
        inserer(liste2,i,n)
        inserer(res,longueur(res)+1,liste2)
      finpour
    finpour
  finsi
  retourner res
fin
```

INSA ROUEN NORMANDIE

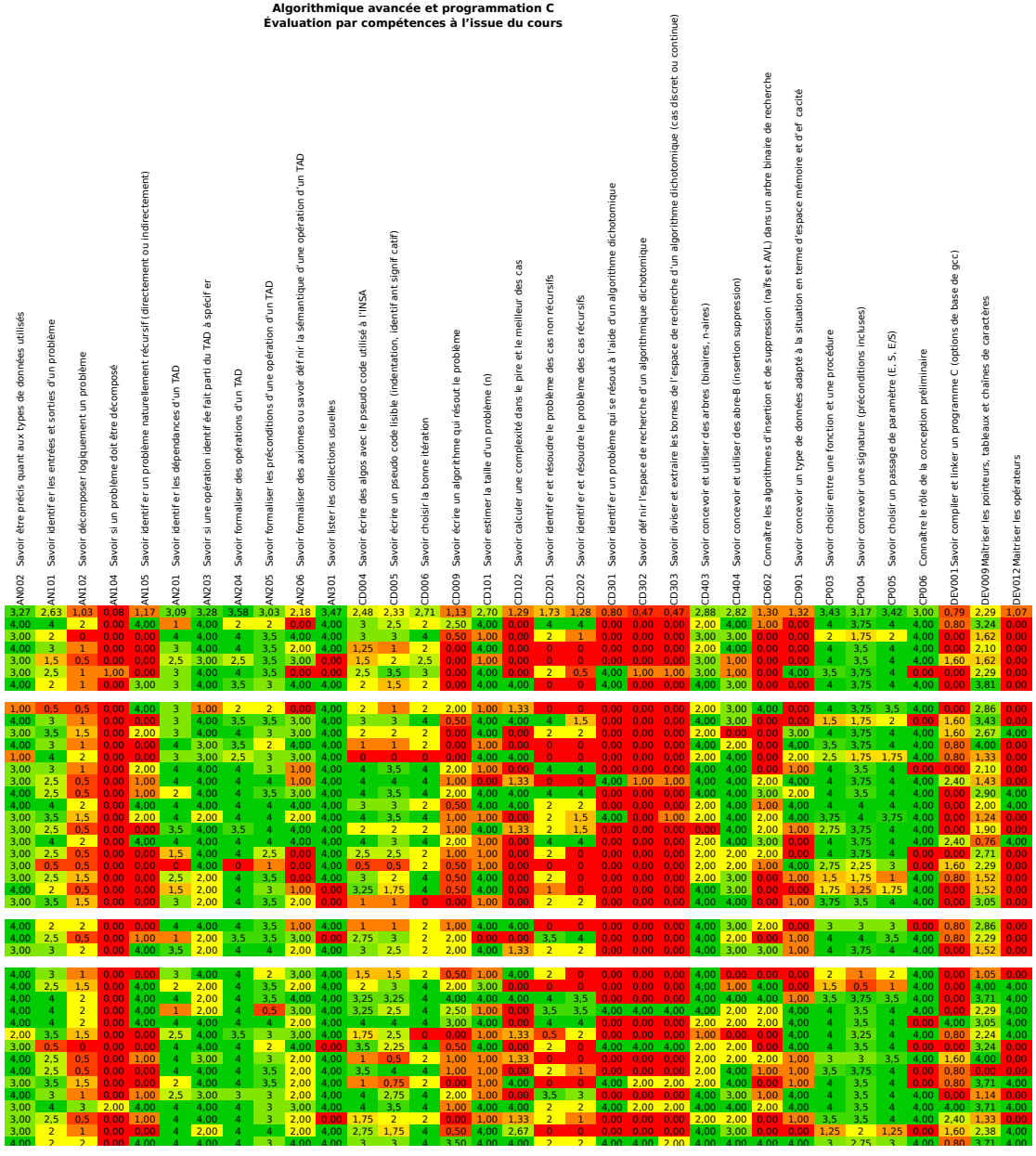
Évaluation 1 / 2

Construction des tableaux de bord

- Lors de la correction d'une question, l'enseignant attribut une « note » pour chaque compétence :
  - 0 pour : la compétence n'est pas du tout acquise ou n'est pas présente sur la copie ;
  - 1 pour : la compétence est utilisée mais avec beaucoup d'erreurs ;
  - 2 pour : la compétence est utilisée avec quelques erreurs ;
  - 3 pour : la compétence est utilisée avec peu d'erreurs ;
  - 4 pour : la compétence est totalement acquise, aucune erreur.
- Pour une copie donnée, on associe donc une note (moyenne pondérée) à chaque compétence
- Tableau de bord = les notes pour chaque compétence et chaque étudiant (0 rouge, 4 vert)
- Deux tableaux de bord produits

Évaluation 2 / 2

Algorithmique avancée et programmation C
Évaluation par compétences à l'issue du cours



Résultats pour l'année 2018-2019 1 / 2

Nombre de compétences évaluées

- 1er examen sur table : 23/70
- 2nd examen sur tableau : 23/70 (coïncidence)
- En tout : 33/70

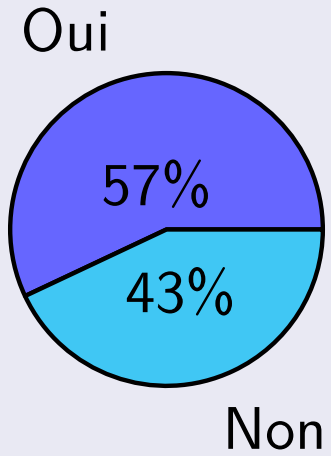
Avantages

- Pour l'équipe enseignante : à l'issue du 1er examen sur table, deux groupes de compétences ont été identifiés comme posant problème (théoriquement prérequis à ce cours)
- Pour les étudiants :
  - difficile de se prononcer car aucun retour dans l'évaluation semestrielle des actes pédagogiques
  - mais aucun étudiant n'a demandé à voir sa copie

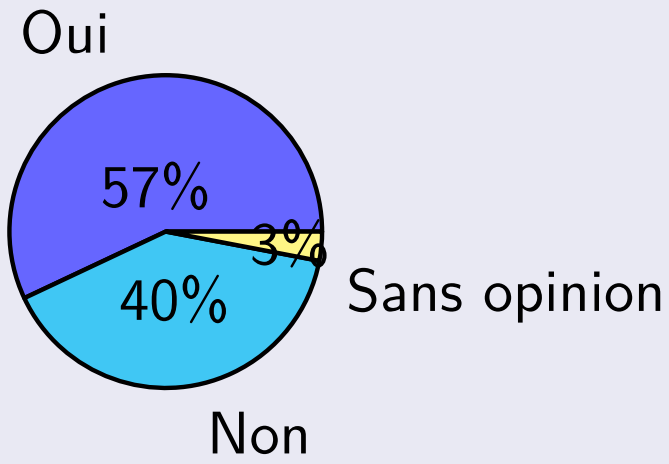
Résultats pour l'année 2018-2019 2 / 2

Enquête

- « Pensez-vous que l'affichage du tableau de bord d'évaluation par compétences a eu une influence sur votre besoin de consulter votre copie ? »



- « Pour un examen futur, pensez-vous que l'affichage de votre évaluation par compétences puisse avoir une influence sur votre besoin de consulter votre copie ? »



Évolutions pour l'année 2019-2020

- Revoir les supports de cours et le livret d'exercices de TD
- Moins de recouvrement entre les compétences évaluées lors du 1er et du 2nd examen
- Développement d'un logiciel ad'hoc

INSA ROUEN NORMANDIE

Merci pour votre attention

Questions ?

INSA
ROUEN NORMANDIE