

Améliorer le retour aux étudiants par des tests unitaires générés à partir de motifs trouvés dans le code de leurs programmes

Julien Liénard¹[0009–0009–2966–7179], Kim Mens¹[0000–0003–0303–1630], and
Siegfried Nijssen¹[0000–0003–2678–1266]

UCLouvain, ICTEAM, Louvain-la-Neuve, Belgium

Abstract. Les premiers cours de programmation à l’université comprenant de nombreux étudiants utilisent couramment des correcteurs automatiques pour fournir aux étudiants des corrections et des retours automatisés sur les exercices de programmation de base. Implémenter un tel retour pour l’intégrer dans un correcteur automatique est une tâche chronophage pour les enseignants. De plus, ce retour est souvent biaisé par la perception de l’enseignant des erreurs que les étudiants pourraient commettre, plutôt que sur les erreurs qu’ils commettent réellement. Nous présentons un prototype d’outil et d’une méthodologie de soutien pour résoudre ce problème. La méthodologie commence par l’analyse du code des programmes soumis par les étudiants des années précédentes, afin d’identifier des motifs récurrents dans leurs solutions. Nous classifions ces motifs en fonction des scores des étudiants: est-ce qu’ils ont réussi ou raté l’exercice? Ensuite, notre outil génère automatiquement des tests unitaires pour ces motifs qui correspondent typiquement aux mauvaises pratiques, aux failles ou aux anomalies observés dans le code des étudiants. Ces tests générés peuvent facilement être intégrés dans un correcteur automatique afin de donner un retour sur ces mauvaises pratiques observés.

Keywords: Enseignement de l’informatique · Minage de motif · Test unitaire · Génération de code · Correcteur et retour automatique