

# L'usage d'un micro-monde pour apprendre la programmation en LOGO

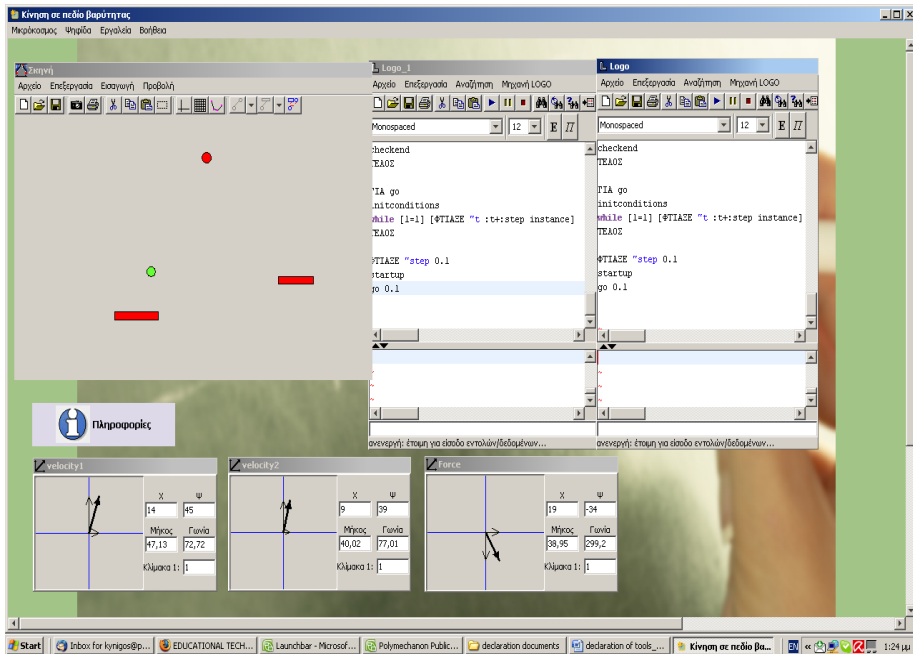
Marios Xenos<sup>1</sup>, Zacharoula Smyrnaïou<sup>1</sup>  
mariosxenos@gmail.com, zsmyrnaiou@ppp.uoa.gr

<sup>1</sup>Laboratoire de technologie de l'éducation,  
Faculté de Philosophie, NK Université d'Athènes  
Ilissia 15784, Athènes, Grèce

L'introduction à la programmation est un processus très difficile pour les élèves. Quand on enseigne des cours d'introduction à la programmation aux élèves, on utilise, le plus souvent, un langage de programmation conventionnel. Selon Brusilovsky et al. (1997) ces langages sont trop vastes et trop spécifiques. Au lieu de souligner les principes fondamentaux, ces langages évoquent des notions secondaires qui reflètent les subtilités du langage donné et sa mise en œuvre. En plus, l'enseignement du langage de programmation est loin de l'expérience des élèves. Un environnement technologique, comme le micromonde *Jongleur* pourrait répondre à ces difficultés, en utilisant le langage LOGO pour programmer le comportement d'objets dans un contexte de problèmes réels et ludiques.

Cet article présente de manière succincte les données recueillies en Grèce dans le cadre d'un projet européen appelé METAFORA et vise à explorer comment le micromonde *Jongleur* permet aux élèves de changer le code en LOGO pour modifier les éléments graphiques et affecter le comportement d'un environnement informatique qui simule le jeu du jonglage. Les élèves ont simultanément deux rôles: celui du concepteur et celui du joueur. Le changement de rôle en conjonction avec les représentations fournies par l'environnement aide les élèves à apprendre les concepts de programmation en LOGO et offre une valeur pédagogique supplémentaire dans l'initiation des élèves à la pensée algorithmique.

Les situations d'enseignement de programmation qui sont étudiées dans le cadre de la recherche utilisent un micromonde spécifique appelé *Jongleur* (figure 1). Sa spécificité consiste à être « demi-cuit » c'est-à-dire « incomplet ». Les micromondes (Kynigos, 2007) qui sont incomplets par leur conception, peuvent mettre les élèves devant le défi de les déconstruire (afin de savoir pourquoi les choses ne fonctionnent pas comme ils le feraient normalement), changer leur code de programmation (en LOGO), les modifier, mettre à profit leurs parties et de créer des artefacts peut-être tout à fait différents que ceux d'origine. Dans ce sens, le micromonde *Jongleur* s'inscrit dans le courant constructionniste de l'apprentissage, conçu pour provoquer les élèves à déconstruire et à (re) construire le modèle qui sous-tend le jeu Jonglage, en leur donnant la possibilité de s'engager dans des activités de production qui comprennent la programmation en LOGO.



**Figure 1 : L'interface du micromonde "jongleur"**

Les élèves effectuant des différentes étapes de l'expérimentation, la déconstruction et la ré-construction du jongleur sont capables d'identifier les composantes du programme et d'apprécier les résultats qui apportent ces modifications.

## Bibliographie

- Ainsworth S., Bibby P., Wood D., (1997) Information technology and multiple representations: new opportunities – new problems *Journal of Information Technology for Teacher Education*, Vol. 6, No. 1, 1997
- Brusilovsky, P., Calabrese, E., Hvorecky, J., Kouchnirenko, A., and Miller, P. (1997) Mini-languages: A Way to Learn Programming Principles. *Education and Information Technologies* 2 (1), pp. 65-83
- Kynigos, C. (2007). Half-Baked Logo Microworlds as Boundary Objects in Integrated Design. *Informatics in Education*, 6 (2), 335–359.
- Smyrniou, Z. & Dimitracopoulou A. (2007). Inquiry learning using a technology-based learning environment. In (Ed) C. Constantinou & Z. Zacharia, *Computer Based Learning in Sciences* (pp. 90-100). Proceedings of 8th International Conference on Computer Based Learning, Heraklion, Crete.