Novice Programmers in the First Three Semesters: Gaining a Better Understanding of the Problem

Malcolm Corney

ALTA Forum, Brisbane
12-13 April, 2012
Project Team

- **QUT**
  - Malcolm Corney
  - Colin Fidge
  - James Hogan
  - Mike Roggenkamp
  - Donna Teague (PhD Student)
- **University of Technology Sydney**
  - Raymond Lister
- **University of Western Australia**
  - Luigi Barone
  - Rachel Cardell-Oliver
Project Team

- All participants are/were teaching 1st, 2nd and 3rd semester programming units
  - Teaching assignments changed
  - Not all of project team now involved
  - Not all replacements keen to participate
What Do We Know?

• Teaching programming is not easy
  – Languages
  – Tools
  – Objects early / Objects late
  – Teachers, engagement, peer learning

• Students learn at different rates
Motivation

- Plethora of literature discussing ways of improving L&T for novice programmers
- Little real data on novice programmers' capabilities
- Most studies concentrate on first programming subject
Project Aims

• **Goal**
  – Longitudinal study (three semesters) of students' performance on test and exam questions

• **Expected Outcomes**
  – An archive of "in class test" and exam questions
  – An anonymised repository of students' tests, final exam scripts etc.
  – Performance data from students from multiple universities
Project Methodology

- Action research approach
  - Two iterations
    - Iteration 1 - Semester 2, 2011
      - Informed by pilot study
      - QUT - CS0, CS1
      - UTS - CS1
    - Iteration 2 - Semester 1, 2012
      - QUT - CS0, CS1, CS2
      - UTS - CS1
      - UWA - CS1
Instruments

• In-Class Tests
  – Not for marks
  – Presented as informal quizzes in lecture sessions
  – Learning opportunity - answers are modelled
  – Gives regular snapshots of understanding of concepts

• Exam Questions
  – Questions similar to in-class tests
  – In longitudinal chain of units
Pilot Study

- CS0
  - Swapping
  - Assignment Statements
  - Reversing
- CS1
  - Explain in Plain English questions
- CS2
  - Soloway's Rainfall Question
Iteration 1 - Semester 2, 2011

• Neo-Piagetian Concepts
  – Conservation
  – Reversibility
  – Transitive Inference
Conservation

- Find the longest string value in an array/list
- Two implementations – choose which option for each line

```python
best = 
  a) 0
  b) names[0]

for index in range(len(names)):
  if (len(names[index]) > 
   a) len(best):
    b) len(names[best]):
      best = 
        a) names(index)
        b) index

print 
  a) names(best)
  b) best
```
Reversibility

• Given code to shift elements of an array one position to the right, write code to shift them back

```python
temp = values[len(values) - 1]
for index in range(len(values) - 1, 0, -1):
    values[index] = values[index - 1]
values[0] = temp
```
Transitive Inference

- Code given that prints smaller of two stored values

```python
if adam < bob:
    print adam
else:
    print bob
```
Transitive Inference

• Code given with three variables

```python
if adam < bob:
    # code to swap adam and bob:
    temp = adam
    adam = bob
    bob = temp
if bob < charlie:
    # assume code to swap bob and charlie is here
if adam < bob:
    # assume code to swap adam and bob is here
```

• Ask for purpose of code in plain English
  – Puts values in descending order
Iteration 2 - Semester 1, 2012

- Leading Questions
- In class tests running each two weeks in first unit
- Will be followed with the NP concept questions in exam
- Progressive questions will be asked in second unit exam
- Progressive questions will be asked in in-class tests in third unit
Outcomes to Date

• Pilot Study
  – 1 x ACE 2011 Paper
  – 2 x ACE 2012 Papers - Best Paper Award
  – After grant was received, this project became part of a successful ALTC grant application - BABELnot
  – ACE 2011 work replicated by Murphy et al, SIGCSE 2012
• Iteration 1
  – Paper submitted for ITiCSE 2012
  – Paper to be submitted to ICER 2012
• Anonymised repository nearing prototype stage
• Informed Improvements for Teaching
Future Work

• Analysis of Iteration 2
• Iteration 3
• More Collaborators
• Acknowledgements
• Questions