ALTA Learning & Teaching grant report

Investigating the nature and design of computer programming examinations

Judy Sheard, *Monash University*
Simon, *University of Newcastle*
Angela Carbone, *Monash University*
Daryl D'Sousa, *RMIT University*
James Harland, *RMIT University*
Raymond Lister, *University of Technology, Sydney*

Motivation

- Formal examination - a common technique for summative assessment of computer programming students.
- Designing an exam paper is an important task.
- … and yet there are very few studies of exam instruments. *How are introductory programming exams constructed? What type of questions are used? What is the pedagogical basis of the design?*
- If we do not understand the instrument we are using, how can interpret the results with any confidence?

Project aims and outcomes

To investigate the *nature* and *composition* of formal examination instruments for introductory programming *and* the *pedagogical intentions* of the educators who construct these instruments.

Potential outcomes:

- find exam questions that are commonly used and could be shared.
- gain ideas for novel exam questions.
- gain an understanding of current practice in exam design.
- a searchable, shareable repository of questions

What we have done …

Prior to ALTA grant:

- Exam question classification scheme developed - characteristics of introductory programming exams
- Workshop at ACE 2011
- 20 exam papers analysed by 12 academics – individual then paired classifications – inter-rater reliability checks.

For the ALTA grant:

- Extension and refinement of classification scheme - complexity of questions
- Workshop at ACE 2012
- 20 exam papers analysed. (ICER 2012 paper in preparation).
- Exploring the pedagogical basis of exams: 12 academics interviewed. (analysis in progress, publication planned for the CSE journal)

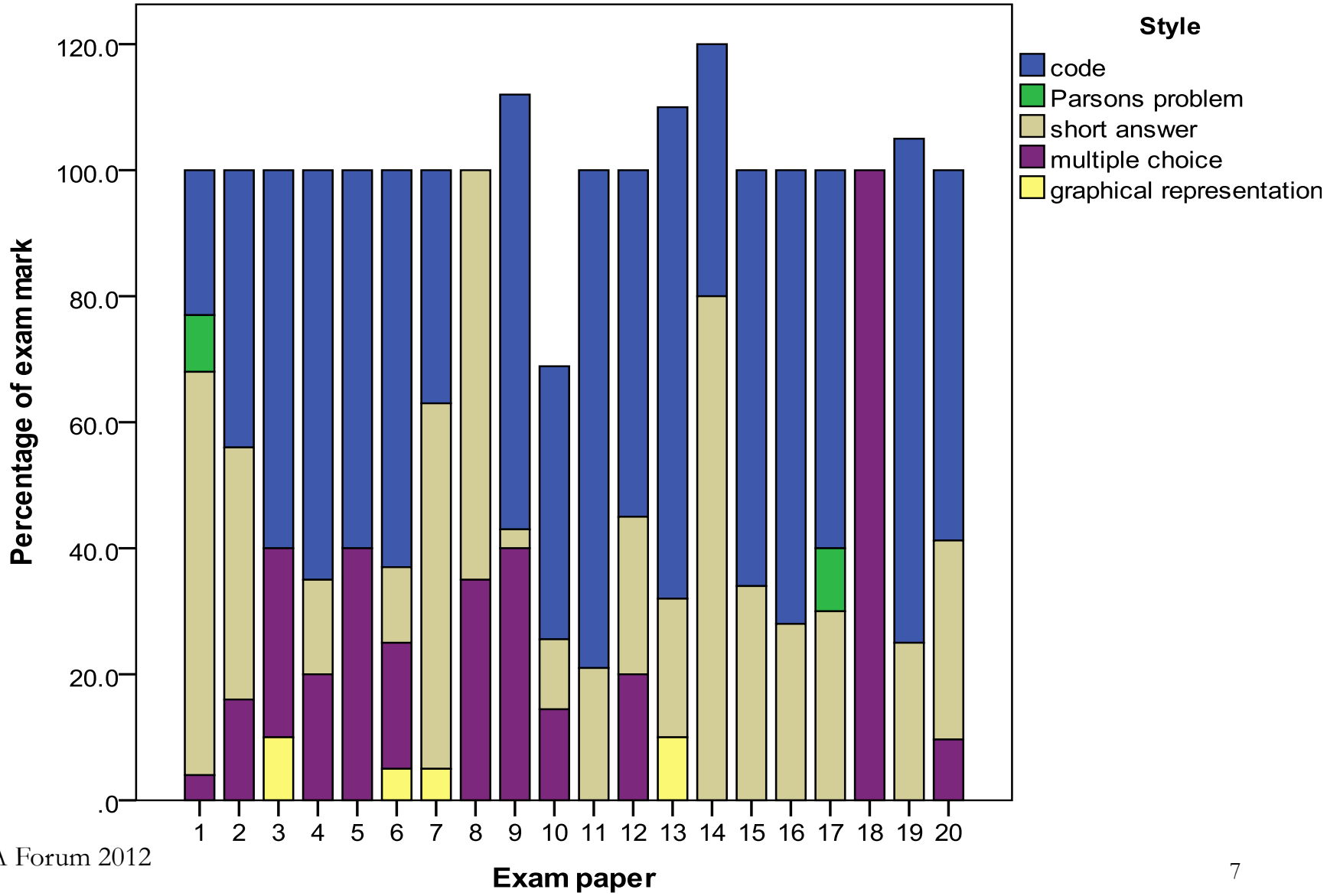
Classification scheme for exam questions

- Percentage mark allocated
- Topics covered by question
- Skill required to answer question
- Style of question
- Open or Closed question
- Measures of complexity:
  - External domain references
  - Explicitness
  - Linguistic complexity
  - Conceptual complexity
  - Intellectual complexity
  - Code length
- Degree of difficulty

Classification scheme for exam questions

- Percentage mark allocated
- Topics covered by question
- Skill required to answer question
- Style of question
- Open or Closed question
- Measures of complexity::
  - External domain references
  - Explicitness
  - Linguistic complexity
  - Conceptual complexity
  - Intellectual complexity
  - Code length
- Degree of difficulty

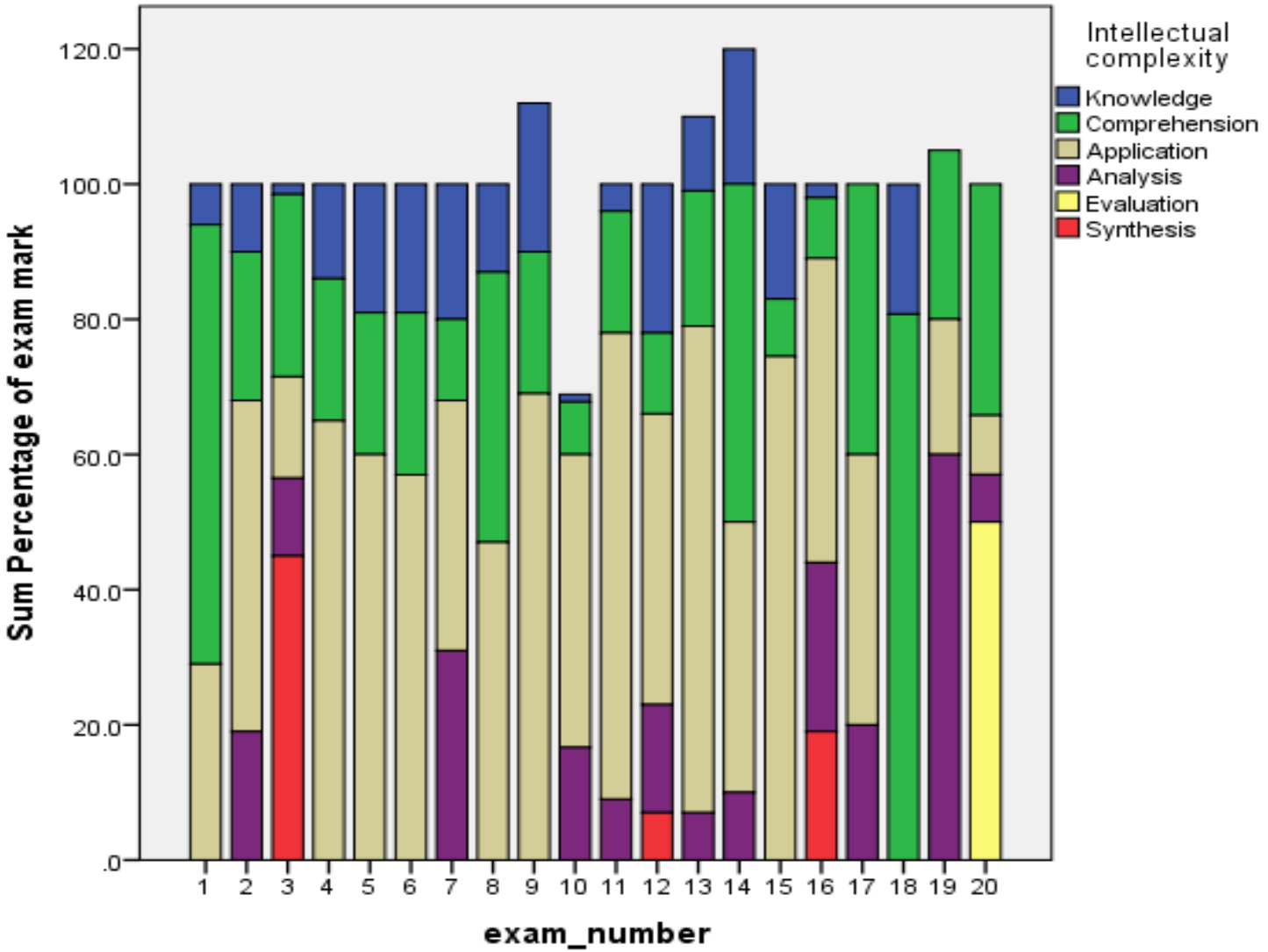
Question style



Classification scheme for exam questions

- Percentage mark allocated
- Topics covered by question
- Skill required to answer question
- Style of question
- Open or Closed question
- Measures of complexity:
  - External domain references
  - Explicitness
  - Linguistic complexity
  - Conceptual complexity
  - Intellectual complexity
  - Code length
- Degree of difficulty

Intellectual complexity



References

[1] Sheard, J., Simon, A. Carbone, D. Chinn, M. Laakso, T. Clear, M. de Raadt, D. D'Souza, D. Harland, R. Lister, A. Philpott, and G. Warburton, "Exploring instruments used to measure programming students' learning: An exam classification scheme," in *ICER 2011*, Providence, Rhode Island, USA, , 2011.

[2] Simon, J. Sheard, A. Carbone, D. Chinn, M.-J. Laakso, T. Clear, M. de Raadt, D. D'Souza, R. Lister, A. Philpott, J. Skene, and G. Warburton, "Introductory programming: Examining the exams," in *14th Australasian Computing Education conference*, Melbourne, Australia, 2012.

Thank you to ALTA for funding this project