An Investigation of Potential Success Factors for an Introductory Model-Driven Programming Course

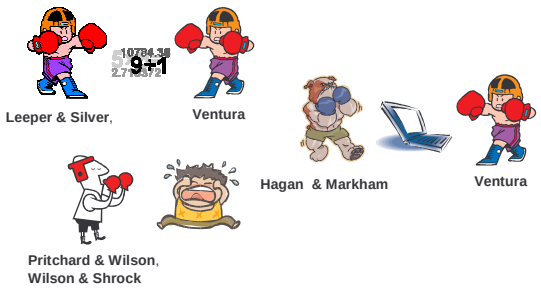
Jens Bennedsen,
IT University West,
Denmark
jbb@it-vest.dk

Michael Caspersen,
Dept. of Computer Science
University of Aarhus,
Denmark
mec@daimi.au.dk

Goal

- Improve the course design of a CS1 model-driven programming course.
- Improve the knowledge of the success of object-first CS1 courses and compare it to "traditional" CS1 courses
- Find new factors
- Not to predict the grade

Previous Research



Results

- **Math** and **course activity** has a positive correlation with the exam score
- **Gender**, **intended major**, **years at the university**, **team** had no correlation with the exam score

Success

- Grade at the final exam
  - Normally binary pass/fail
  - We post-marked the exams
    - Scale: 00, 03, 5, 6, 7, 8, 9, 10, 11, 13
    - Six or above means pass -> all post marked exercises with six or above did also pass
- No other grades count
  - No assignment or other during the course

Model driven CS1 course

- First half of a CS1 course
  - 7 weeks
- Goals
  - use conceptual modelling in relation to preparing simple object-oriented programs,
  - implement simple OO-models in a modern programming language,
  - use fundamental elements in a modern programming language,
  - use selected class libraries.
- Progression according to complexity of the underlying class model.

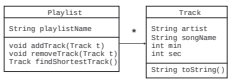
Exam

- 20 students are tested concurrently.
- The test takes place in a lab.
- Nine small progressive programming tasks,
- 1 hour in total, approx. 30 min of actual programming
- All exercises are instances of the same generic exercise

Exam - example

1. Create a class, *Track*, that represents a piece of music; the *Track* class is specified in the following UML diagram:

2. The four field variables must be initialized in a constructor (through four parameters of suitable types). The method *toString* must return a string representation for a piece of music, e.g.
   "Yesterday: The Beatles (2:05)"
3. Create a test method named *exam* in class *Driver*. The method must be static, have return type void, and have no parameters.
4. Create two *Track* objects ,,,
5. Create a new class, *Playlist*, representing a collection of *Track*s; the *Playlist* class and its relation to the *Track* class is specified in the following UML diagram:



Research questions

- What is the relationship of **math**ematical ability to model-based CS1?
- What is the relationship of **gender** to model-based CS1?
- What is the relationship of **major/intended major** to model-based CS1?
- What is the relationship of **course work** to model-based CS1?
- What is the relationship of **years at the university** to model-based CS1?
- What is the relationship of the **team** to model-based CS1?

Method

- Data sources: Adm-sys, TA's, web-site
- Multiple regression analysis
- Start with a model including all variables and interaction variables
  - 34.4% explanation power
  - Eliminate one by one insignificant variables at the 95% confidence interval.
  - End up with a model with two variables explaining 24,4% of the variation of the exam score

The students

- Approximately 235 students
  - from a variety of study programmes
    - computer science,
    - mathematics,
    - geology,
    - nano science,
    - economy,
    - multimedia,
  - 40% are majors in computer science.

Math

- The students score from their high school exam.
  - Three levels: A, B, C – A is required.
- Result
  - the math score from high school has a positive impact on the exam grade
  - math grade alone accounts for over 15% (15,4%) of the variance of the final grade.

Course work

- During the course, the students are required to complete five out of six weekly
- The TA's evaluate the exercises and the score for each exercise is encoded as one of the numbers 1, 2, or 4.
  1. Perfect, no significant errors
  2. OK, small errors
  4. Not accepted/Not handed in
- In case a student got a "4", he had the possibility of resubmitting the exercise once.

Course work - results

- It has a positive impact on the exam grade.
- Course work alone accounts for 7% of the variance of the final grade
  - only half the impact of the math grade from high school.

Study age

- The number of years the student has been enrolled at the university. Integer value from 0 – 20. Students enrolled in 1984 or earlier were coded as 20.
- Not significant neither at the 95% confidence interval
  - 90 %: the variable is not significant in itself but in combination with the math grade, it has a positive impact; with course work, it has a negative impact. These two combinations of variables accounts for 2% of the variation each

Gender

- It is not significant, neither at the 95% confidence interval nor at the 90% confidence interval.
- We can therefore not accept the hypothesis that gender has an impact on the exam score.

Major

- 4 variables: COMPSCIENCE, GEOLOGY, MATHEMATICS, NANOSCIENCE
- None are significant at the 95 % conf. interval
- We can reject the hypothesis about students majoring in comp. science perform better.
- 90% confidence interval: COMPSCIENCE and GEOLOGY were significant.
  - At this level we can accept the hypothesis of a positive impact of majoring in computer science (it accounts for 3,6% of the variance),
  - GEOLOGY is significant, NANOSCIENCE and MATHEMATICS are not → we can not say anything about the students not majoring in computer science.

Team

- There is an a priori correlation between team and intended major because of the way students are allocated to teams
- Since intended major is not significant, the same is true for team.

Regression formula

- GRADE = 1.118 +
  0.589*MATH +
  0.341*COURSEWORK

A number between 0 and 13

Variable	Unstandardised coefficients		Significance
	B	Std. Error	
COURSEWORK	0.341	0.097	0.000
MATH	0.589	0.107	0.001

Discussion

- Math L
- Course work J
- 24,4 % explanation power => lot of other factors
- Conclusions: Hard to give since it is based on our course (and so is all the other studies)

Future work

- Motivation.
- Effort.
- Power of abstraction.
- Prior programming experience.
- Emotional health/well-being

Ad

- Research on pedagogical patterns – please take survey sent to the SIGCSE mailing list
  - If you are not on the mailing list let me know