

# The COMP1120 Experiment in 'Fast-track' First Year CS Teaching

*or*

*And then there were twelve.*

Peter Strazdins,  
Department of Computer Science,  
Australian National University,

<http://cs.anu.edu.au/~Peter.Strazdins/seminars/comp1120>

5 Nov 2003

# 1 Talk Outline

- overview and objectives of COMP1120
- organisation of COMP1120 in 2003
- some details in running the course
- entry into COMP1120
- course results and their analysis
- strengths and weaknesses of the COMP1120 model
- suggestions for improvement
- other models for achieving the same objectives
- conclusions

## 2 Objectives and Overview of COMP1120

- we now have a diverse range of experience (and potential ability) in our 1st year CS intake, who do the following courses:
  1. COMP1100: Introduction to Programming and Algorithms
  2. COMP1110: Foundations of Software Engineering
- (anecdotally) the more experienced students:
  - got bored with the slow pace, esp. of COMP1100
  - often sidetracked tutors & intimidated less experienced students
  - a 2002 survey indicated about 50 COMP1100 students regarded themselves as 'highly experienced'
- **idea:** identify these and put them in a combined 1 semester course:
  - COMP1120: From Programming to Software Engineering
- to solve these problems
- motivations include separating these students and giving them something more challenging (Honours potential?)
  - see entry web page for elaboration

### 3 Organization of COMP1120

- COMP1120 incorporated the Curriculum Development Workshop 12/02 proposed changes to COMP1100 & COMP1110
- course divided into 3 sequential modules:
  1. IP: *Introductory Programming* (COMP1100), weeks 2–6, Richard Walker & Malcolm Newey
  2. DA: *Data Structures and Algorithms* (COMP1110), weeks 7–9, Peter Strazdins
  3. SE: *Software Engineering* (COMP1110), weeks 9–13, Carol Edmondson
- had 31 lectures and 12 2-hour combined tutorial/laboratories (tutor John Uhlmann)
- one assignment per module; DA & SE assignments based on a UBoat simulation theme:
  - high potential for illustrating OO concepts, highly challenging and fun!

## 4 Entry into COMP1120

- promotion (PEA day, O-week lect, COMP1100 lect 1 & 4)
  - with web page for more info
- weeks 0–1: prepare for application and entry Quiz
- Friday week 1: introductory lecture & Quiz (19 applicants)
  - had repechage Quiz in early week 2 (6 more applicants)
  - informed the 18 successful applicants by end of week 2
- week 7: notification of formal permission to enrol
  - based on IP module's programming assignment (due mid week 6), and a 2 hour mid-semester exam (end week 6)
  - effectively withdraw from COMP1100 and enrol in COMP1120
  - 'drop-back': 4 candidates did not reach this stage; 2 then opted out
- week 11–12: ANU system actually accepted the enrolments!
- had considerable staff workload implications (lucky not > 50 applicants!)

## 5 Status of COMP1120 Materials and Workload Issues

- in principle, all materials already in those of COMP1100 and COMP1110
    - compression and re-organization factors however made preparation significant
    - COMP1110 materials: overly elaborate & have too many internal constraints; however excellently archived!
  - sources for most lectures, and all other documents, similarly archived
  - providing *interesting/challenging* practical work took considerable extra effort
    - chose UBoat simulation theme, used in sem. 1 2nd year, 1993–5
    - ‘port’ (of large complex system) from Modula-2 to Eiffel was no joke:
      - cumbersome for calculations, too strictly OO, some bits were beyond current DCS experience, no co-routine support
      - *don't anyone try to tell me that Eiffel has no gotchas!*
      - old X-windows code is now broken; only partially fixed
  - in general, a course with 2 courses material has more than one course's workload
- (may be better 2nd time though...)

## 6 COMP1120 results

- COMP1120 students:

degree	PhB	PhB	LLB	PhB	BIT	BSE	BIT	BSc	BSE	BIT	BSE	BSE
UAI	99.7	99.7	99.9	99.3	98.5	97.2		96.6	95.8	93.3	88.1	
Quiz	20	3	24	16	16	22	20	15	7	21	14	7
Prog Exp	4	5	5	2	2	3	4	4	5	2	5	3
	Per	Jav	C	Pas	VB	VB	C+	Ftn	Eif	VB	C+	C
'1100'	90	76	89	83	82	83	74	79	72	69	72	58
1120	95	89	87	85	81	74	76	70	63	61	50	31

- drop-back COMP1100 / not offered COMP1120

degree	BSE	BSE	BSE	BSE	BSE	BIT	LLB	BIT	BSE	PhB	BE	BSE	BIT
UAI		99.9	94.3				95.2		97.7	99.6			
Quiz	15	15	15	14	19	13	1	7	4	1	0	3	6
Prog Exp	4	1	3	1	3	4	2	2	1	1	1	3	1
	Jav	C	Pas	VB	Pas	C+	Jav	C+	VB	Pas	Jav	C+	C
'1100'			54	52									
1100	94	93	92	92	55	DA	89	86	84	79	67	59	55

- all were male !*

## 7 Analysis of COMP1120 results

- UAI: quite a good predictor of final marks
- Quiz: week 1–2: predict final values in (tricky) SALTY code
  - tested narrow and specific skills; could be broadened
- Programming Experience: based on description + code samples (if any); subjective ratings  $\leq 3$  often had voluminous but shallow code
- '1100' result: (week 6–7): based on 1st assignment & 2 hour exam
  - several students clearly had trouble with the fast pace
- could maybe have given another 4 offers (promising UAI)
- note that 1120 results and 1100 results are not directly comparable
  - many students lost marks in the SE exam questions
- COMP1120 would have made an IT degree at ANU more attractive:
  - 0.52 (self), 0.73 (acquaintances), 1.43 (in general)
  - 2 = 'strongly agree', 1 = 'weakly agree', 0 = 'neutral'
  - 1 student comments: "most of my friends went to UCan because [their degree] is more fast-track"

## 8 Strengths and weaknesses of the COMP1120 model

- ✓ students got privileged, interesting tuition; high satisfaction ratings
  - Richard's OO expertise, tutor & assignment theme got good raves!
- ✓ assessment issues easier with a separate course
- ✓ opportunities to pursue another course (BSEng students can do a proper major)
- ✓/× easy drop-back into COMP1100
- × few enrolments despite efforts; possible reasons include:
  - looked too daunting, too fast-track;  
much risk and extra effort to get same marks; timetable clashes
- × high staff resources required
- × crossing enrolment boundaries proved to be a significant barrier ...
  - also had problems in enrolling in COMP2600 in semester 2
- × no follow-up for the next 5 semesters ...

## 9 Suggestions for Improvement

- entry into course: make more light-weight, 'at-your-own-risk'
  - still have a (diversified SALTY Quiz, but use results in a more advisory way
- IP module too fast for some (esp. if no prior OO experience)
  - could be at a slightly slower pace, more emphasis on the basics
  - also could be more structured
- UBoat codes could have a better (more 'OO correct') design
- find some way of emphasising the importance of SE

## 10 Alternative Models

- 0-unit crash-course in Programming in Eiffel (or current intro. language)
  - ✓ less expensive; could be run in both semesters (facilitates mid-year entry, transfers from other unis)
  - ✗ loses *more* EFTSUs; timescale would be very problematic
- Advanced Stream in COMP1100/10 (and perhaps beyond)
  - need not attend lectures if find boring
  - separate tutorials/labs: quickly get through basics, then do more advanced exercises / assignments
    - ✗ still some extra effort, and some problems in equating assessment
  - ✓ more flexible, and more sustainable; more potential for follow-up

## 11 Conclusions

- despite the low enrolments, COMP1120 ran successfully
  - the team of 4 lecturers worked, but fewer maybe more efficient
  - most materials are well-developed
- are we getting any marketing advantages from COMP1120?
  - is there any real economic payoff for running it?
- should we continue to run it?
  - IMHO no: not cost-effective
    - feel unlikely that *completions* could get to 35–40: simply not attractive enough !
  - but if DCS insists, let me do it! (it was fun!)
    - and the UBoats are seaworthy for at least another year...
- if not, should we seek an alternate model?
  - feel the priority is to consolidate the mainstream (COMP1100/10) first

Or consolidate our resources elsewhere?

- (sorry, this seminar has fewer than 16 slides ...)