CS2100 Final Assessment Report

AY2020/21 Semester 2

This is the report for the CS2100 final assessment held on 27 April 2020.

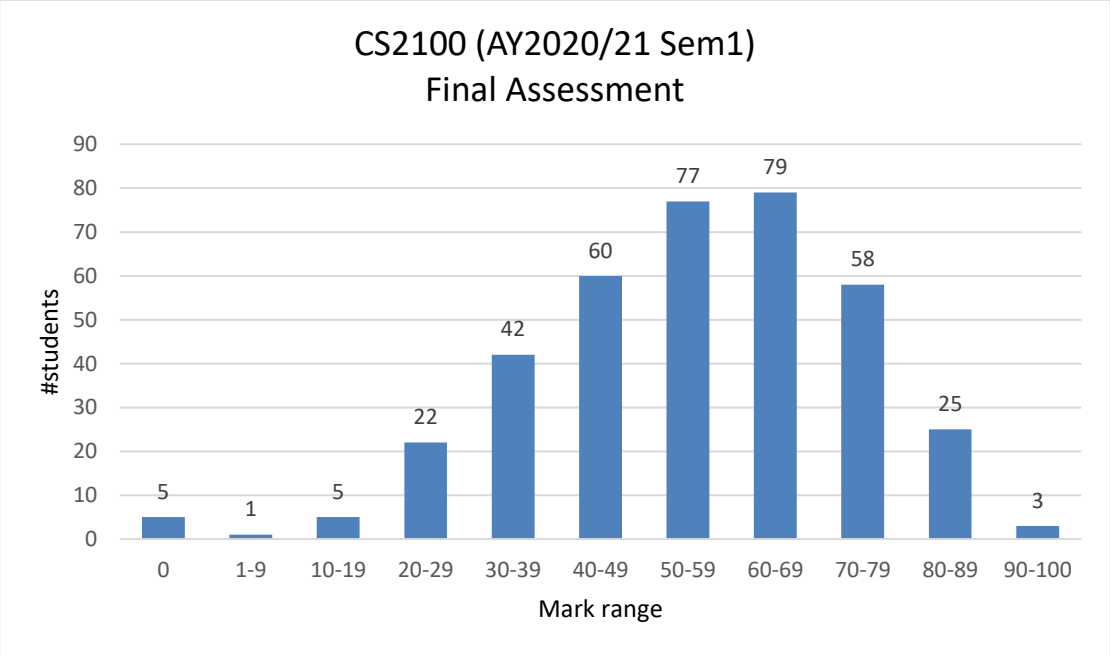
372 (out of 377) students sat for the final assessment.

Below are the statistics:

1. General statistics

	Part A (Q1-6) (12 marks)	Part B (Q7-11) (15 marks)	Q12 (12 marks)	Q13 (13 marks)	Q14 (13 marks)	Q15 (18 marks)	Q16 (14 marks)	**Total (100 marks)**
Average	7.68 (64.0%)	5.71 (38.0%)	8.93 (74.5%)	6.21 (47.8%)	7.20 (55.4%)	9.42 (52.3%)	6.71 (47.9%)	**54.7** (54.7%)
Median	8	6	9	7	8	11	8	56
Standard deviation	2.89	3.49	2.93	4.12	3.68	5.58	4.24	18.17

Below is the chart for the overall results.



Please note that for exams, you are <u>not</u> allowed to write your name anywhere in your file. This is to ensure blind grading. Instruction is given on the first page of the paper, yet a number of students wrote their name in their file.

2. Part A: MCQs 1 - 6

The table below shows the percentage of students who chose the correct answers, and of those who chose the most popular wrong answers:

	Q1	Q2	Q3	Q4	Q5	Q6
	Hardest	Easiest				
%students who chose the correct answer	B 52.0%	C 83.6%	C 58.1%	E 53.3%	C 60.2%	D 76.7%
%students who chose the most popular wrong answer	A 21.8%	D 8.2%	B 17.0%	D 17.0%	D 16.5%	E 10.3%

Interestingly, the hardest and the easiest questions (Q1 and Q2) are on the same K-map. While most students are able to get the number of EPIs correct, many didn't get the number of PIs correct.

For Q3, many students picked option B (108) as the answer, probably missing out counting the one instruction in types *A* and *B* each.

3. Part B: MRQs 7 - 11

The table below shows the answer and average mark for each question:

	Q7	Q8	Q9	Q10	Q11
Answer	A,C,D,E	B,C	A,B,C	C,D,E	A,B,C
Average mark	0.4	1.06	2.1	1.85	0.4
	Hardest		Easiest		Hardest

4. Comments on Q12 (by Colin)

Q12a: Not really a mistake. The edge from state 3 to state 2 is often drawn as two edges (one with x=0 and another with x=1), when a single edge with both labels suffices.

Q12b: Some students called their flip flops *A* and *B* (they should be *F* and *G*), leading to the incorrect equations $DF = A' \cdot x' + A \cdot B$ and $DG = B' \cdot x'$, which are wrong.

5. Comments on Q14 (by Colin)

Q14a: Students often remember to do `la $t0, size` but forget `lw $s2, 0($t0)`.

Q14b: Some students think that doing `la $s0, $s5` makes $s0 = $a5.

Q14c: Most got this either completely right or flat out wrong. Some students do this:

```
count = 0;
i = 0;
while (i < size) {
    if (A[i] % 2 == 1) {
        A[i] = A[i] - B[i];
        i++;
        count++;
    } else {
        i++;
    }
}
```

Which is not quite right because I is incremented in only one place in the original MIPS code, not 2 places. More seriously, some students use variable names like $t0, which are not legal C variable names.

Q14d-f: Students either get this right, or get some small part of the instruction encoding wrong (which makes the entire thing wrong!)

6. Comments on Q15 (by Aaron)

Many students got parts (a) to (e) correct. For parts (f) and (g), some students gave the answers for only one iteration. If the answers for one iteration are correct, partial credits are given.

The question asks for "how many hits in total are there in the instruction cache in the execution of the code". The part about "consider only instructions I1 to I16" is to exclude those instructions outside of these 16 instructions in the original code. Some students were misled into thinking that only one iteration is required.

7. Comments on Q16 (by Aaron)

Partial credits are given for parts (b) – (d) if the student's answer deviates a little from the correct answer. In addition to this, to avoid double penalty, partial credit is also given to part (c) if student's answer for part (c) is 4 less than his/her answer for part (b).

For parts (b) – (d), quite a number of students didn't read the instructions carefully and wrote the total number of cycles of delay, instead of the additional cycles of delay compared to an ideal pipeline (part (a)).

For part (e), there are a number of acceptable answers. However, some of the answers given by students will result in the code becoming incorrect. For move that involves an instruction after a branch to be moved before the branch, student is expected to explain why such a move does not affect the correctness of the code, or full credit will not be given. Also, if the move involves I15, students should explain what would happen to the label skip. Some students also forgot to mention how many cycles of delay are reduced by the move.

Prepared by Aaron Tan

Update: 2 May 2021