note @430 ⌾ ☆   **141** views

[Exams] Past Exams 2016 and older Q&A

Discuss all questions pertaining to exams which took place in 2016 and older here.

You can find the past exams here: https://cs61c.org/resources/exams

When posting questions, you **MUST** reference the semester, exam, **AND** question so we can help you.
Please put this at the beginning of your post in this format: **[{Year} {Semester}-{Exam}]:Q{Question Number}**
For example: **[2016 SP-MT1]:Q1**, or **[2014 SU-F]:Q3**

{Semester} is one of these: SP, SU, FA
{Exam} is of these: Q, MT, MT1, MT2, F
Please separate out parts with periods: 1.2.ii.a.b.3.a

If you follow this format, it will make it very easy to search for similar questions!

midterm1   midterm2   final

~ An instructor (Jerry Xu) thinks this is a good note ~

Updated 27 days ago by Stephan Kaminsky

---

followup discussions *for lingering questions and comments*

◉ Resolved   ○ Unresolved

This was marked a duplicate to the question/note above by Stephan Kaminsky 1 month ago
""

Anonymous Beaker 1 month ago   Spring 2015 MT2 cache operation second part part b 9/16
The question is on guerrilla worksheet 5. It is 5.2 part b. I don't know why the answer is 9/16 instead of 1/2.
Could someone go through it in detail? Thanks!
helpful! │ 0

> **Xxxxxxx** 1 month ago   In the first loop, since the step size of each iteration is $32 bytes$ and the block size is $32 bytes$, every time we access `arr[i]` it will be in a unique cache block. Also, the array size is $64*1024*4 byte = 256KB$, which is 8 times as the size of the cache. So we can conclude that after the first loop the last $1/8$ of the array is in the cache. Now for the second loop, assume the total number of accesses is N, half of them are read and the others are write. Since the cache can only hold $1/8$ of the array at a time, the first $\frac{1}{8}*\frac{1}{2}N*2 = \frac{1}{8}N$ accesses are cache hits. For the other part, there will be misses on read (`arr[i]`) and hits on following write (`arr[i+1]`) after we evict and replace a block with new data. The number of cache hits is $\frac{7}{8}*\frac{1}{2}N = \frac{7}{16}N$. Thus the hit rate is
>
> $$\frac{\frac{1}{8}N + \frac{7}{16}N}{N} = \frac{9}{16}$$
> helpful! │ 0

Anonymous Beaker 1 month ago   Thank you, Zhiying!

Anonymous Scale 1 month ago

5   More Pipeline Hazards (FA16 Q3)

The goal of the problem is to increase the execution speed of the code below by eliminating as many stalls and useless operations (nops in the branch delay slots) as possible. The code runs on a 5- stage pipelined RISC-V CPU with forwarding with the characteristics discussed in lecture.

] Indicate stalls in the code on the next page with arrows right after the instruction where the stall occurs. The following is not necessarily correct and only used as an example:

stall →

| addiu s0 s0 1 |
| addiu s1 s1 4 |
| addiu s2 s2 4 |

stall after first lw, stall after second lw

] How many cycles does it take to execute the entire code sequence below, including stalls?

19 cycles

11 instructions + 4 stall cycles + 4 cycles at the end = 19 cycles 13 instructions + 2 stall cycles + 4 cycles at the end = 19 cycles Note that we add 4 cycles at the end because there are 5 stages and the last instruction must finish all 5 stages and each stage takes 1 cycle. (Same reason why 1 instruction would need 5 cycles to complete.)

This is a question from guerilla section 04.

It seems like the solution is proposing two different ways to achieve 19 cycles.

Which way is the correct one ? How many cycles are we stalling for lw ? How many cycles are we stalling for beq ?

Thanks

helpful! | 0

Sunay Poole 1 month ago   Edward just updated the solutions!

helpful! | 0

Anonymous Scale 1 month ago   Thank you so much, the solution makes sense now.

helpful! | 0