

note @431  

132 views

[Exams] Past Exams 2017 Q&A

Discuss all questions pertaining to exams which took place in 2017 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: **[[Semester]-{Exam}]:Q{Question Number}**
For example: **[SP-MT1]:Q1**, or **[SU-MT2]:Q3**

{Semester} is one of these: SP, SU, FA

{Exam} is of 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

1 endorsed followup comment

Resolved Unresolved



Anonymous Atom 2 months ago

[FA-MT1]:Q4.2 I am wondering why the solution is 20. I thought there is 1 byte for self, 4 bytes for word pointer, 4 bytes each for left, right, and sub pointers in a 32-bit computer architecture.

helpful! | 0



Xxxxxxx 2 months ago On a 32-bit word-aligned architecture, the number of bytes to allocate should be multiples of 4.

The struct takes 17 bytes in total, so it should be padded to 20 bytes.

~ An instructor (Caroline Liu) thinks this is a good comment ~

helpful! | 2

Resolved Unresolved



Anonymous Calc 2 months ago

Rendering markdown...

helpful! | 0



XXXXXXX 2 months ago The address of the head pointer is passed in to `reversed()`. Thus if you deference and modify `head_ptr`, the change would be permanent to the original linked list. The line `*head_ptr = (*head_ptr)->next` basically removed the first node from the original link list, reassigning and losing the original head pointer. The condition for termination of the while loop is that `*head_ptr == NULL`, at that point you've lost the entire original linked list.

The proper way to do this is to set a temporary pointer `struct list_node* current = *head_ptr`. Then use that pointer to traverse the linked list within the loop: `current = current->next`. That way you're changing `current` but not `*head_ptr` itself. When the loop ends `current == NULL` but the original `*head_ptr` is still intact.

helpful! | 1

Resolved Unresolved



Anonymous Poet 2 months ago

For 1,b could you please explain what is the three-digit base-7 number? I guess it is like 666 666. But in this case, why we need 9 bits to represent unsigned 7^5 ? Thx

helpful! | 0



XXXXXXX 2 months ago A three-digit base-7 number would be like $(666)_7$. The question is "What is the minimum number of bits needed to represent all the unsigned integer values that a three-digit base-7 number could encode?"

To find the number of unsigned integer values that a three-digit base-7 number could encode, we can find the value of the largest three-digit base-7 number: $(666)_7 = (6 \cdot 7^2) + (6 \cdot 7^1) + (6 \cdot 7^0) = 342$.

Alternatively, you can observe that this number is the same as $7^3 - 1 = 342$.

To find the number of bits for this number, we just calculate $\lceil \log_2 342 \rceil = 9$. Therefore, you would need 9 bits to represent all unsigned integer values that a three-digit base-7 number could encode.

helpful! | 0

Resolved Unresolved



Anonymous Poet 2 months ago

For 2. 5, char *word1 = "GO"

What type of address does word1 evaluate to? -- answer is static . But word is a variable in function, why is it not in stack?

helpful! | 0



XXXXXXX 2 months ago "GO" is a string literal, and string literals are created in static memory.

helpful! | 0

Resolved Unresolved



XXXXXXX 1 month ago

[FA-MT2] 4.c Just to clarify, in general, is the PC resolved in the EX stage (right after ALU)?

helpful! | 0



XXXXXXX 1 month ago Do you mean the PC for next instruction? I think so.

helpful! | 0



Xxxxxxx 1 month ago yeet

helpful! | 0

Resolved Unresolved



Anonymous Beaker 1 month ago

Rendering markdown...

helpful! | 0



Anonymous Beaker 1 month ago

Rendering markdown...

helpful! | 0



Anonymous Beaker 1 month ago

Rendering markdown...

helpful! | 0



Anonymous Beaker 1 month ago Ignore, I reposted the question with the correct format below.

helpful! | 0

Resolved Unresolved



Anonymous Poet 1 month ago

[FA-MT2]: Q4.2 Q1. e-f-g,

I am a little confused about these questions. How to consider "How many passes would the conventional assembler make" and what does the "assembler now started backwards from the last line of the program" mean?

Could you give some explanation about these questions? Thanks

helpful! | 0



Anonymous Poet 1 month ago Sorry, it should be Q1, not Q4.2

helpful! | 0



Xxxxxxx 1 month ago It just means how the assembler reads the entire assembly file and convert it to an object file for the program. Normally the assembler reads every line from top to bottom. The labels in branch/jump instructions will be replaced by relative offset. But one issue is, it cannot figure out the offset if it's not seen yet in the first pass (for example, TRIX is referenced by jal on line 4 but the code start at line 6). In order to resolve this, it reads the file for second time, and it should figure out all offsets since it already knows where all the labels are in the first pass.

Now, if the assembler reads backwards, in this case it actually solves this issue because both TRIX and JOAK are below where they are referenced. So the assembler can figure out the offsets for bne and jal. But in general, some labels appear before they are referenced while others are opposite so there's no guarantee unless the assembler always takes two passes.

For g, those are the tasks for assembler. I'll suggest you review the lecture.

helpful! | 0

Resolved Unresolved



Anonymous Poet 1 month ago

[FA-MT2]: Q3.3

Why RegWEn in 3.3 is write disabled?

helpful! | 0



Xxxxxxx 1 month ago RegWEn needs to be write disabled. Think about the case when the given instruction is cse and $R[rs1] == R[rs2]$. If you set RegWEn to W, then it will write to both memory and rd. It's a little confusing here because cse is an R-type instruction. The logic for generating RegWEn should be updated accordingly too.

helpful! | 0

Resolved Unresolved



Anonymous Beaker 1 month ago

Rendering markdown...

helpful! | 0



Xxxxxxx 1 month ago jal is the one being expanded. jal offset is a pseudo-instruction for jal rd, offset. Multiple object files are linked together during linking stage (by linker).

helpful! | 1

Resolved Unresolved



Anonymous Atom 29 days ago

[FA-Final]: Q13.4d Why is it that small random reads have higher throughput on RAID5 than on RAID1?

helpful! | 0



Sunay Poole 27 days ago We can parallelize the reading across the disks instead of only reading from one disk at a time!

helpful! | 0

Resolved Unresolved



Anonymous Scale 27 days ago

[FA-FINAL] Q.10

2. Assume cache loading/reloading is slower than thread switching. Which is the slowest correct function if:
a. the size of the cache block is equal to $4 * \text{sizeof}(\text{int})$?
b. the size of the cache block is equal to $\text{sizeof}(\text{int})$?

I don't really understand how to solve the problem, hope to get some advice.

helpful! | 0