Want to go back to the demo homescreen? **Click here**

note @433 ☆ **210** views

[Exams] Past Exams 2019 Q&A

Discuss all questions pertaining to exams which took place in 2019 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!

Spring 2019 Final Reference Sheet: https://docs.google.com/document/d/1dE8dah-jxFv1v2HgtSg1zr1VjcmykhPJz6CpqS0dN9U/edit
Summer 2019 Final Reference Sheet: https://drive.google.com/file/d/1RLGSPKCfZUVFOKZrBm_B7ay0xC1d1I6W/view

Other Resources:

- You can check out the video walkthroughs for sp19-mt1 made by a lovely past tutor Evan Sum here: https://www.youtube.com/watch?v=ZxMfbSXJS0U&list=PLXdI-bKzbrT5qIMUdt3_M8FA764eIlWdG
- Here's a video walkthrough by Daniel for the Sp19 Final: https://www.youtube.com/watch?v=8DiN5Hu9x24&list=PLDoI-XvXO0apuEacxuUrUaBq2YDuKYPtV&index=2 (handout and timestamps in comments)
- Here's a video walkthrough for the Su19 Final made by Sunay. You may have to adjust the quality manually.
  - Q1 Potpourri: https://youtu.be/FY5dAMrXvxo
  - Q2 FSM: https://youtu.be/gmHbw6LSeSw
  - Q3 C Coding: https://youtu.be/v4B1WTs5UNU
  - Q4 RISC-V: https://youtu.be/2VHjG-gy9Dk
  - Q5 Data-Level Parallelism: https://youtu.be/oG9Rrzmi0M4
  - Q6 RAID and ECC: https://youtu.be/rfCNTIzNZ2M
  - Q7 Caches: https://youtu.be/xojc8YZaO3Q
  - Q8 Spark: https://youtu.be/A37BFXRXmm0
  - Q9 Datapath: https://youtu.be/q-T4N3hBhUM
  - Q10 Digital Logic: https://youtu.be/3RI36IsDSg4
  - Q11 Virtual Memory: https://youtu.be/5_2fKsK4I34

midterm1 midterm2 final

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

Updated 27 days ago by Stephan Kaminsky and Sunay Poole

followup discussions *for lingering questions and comments*

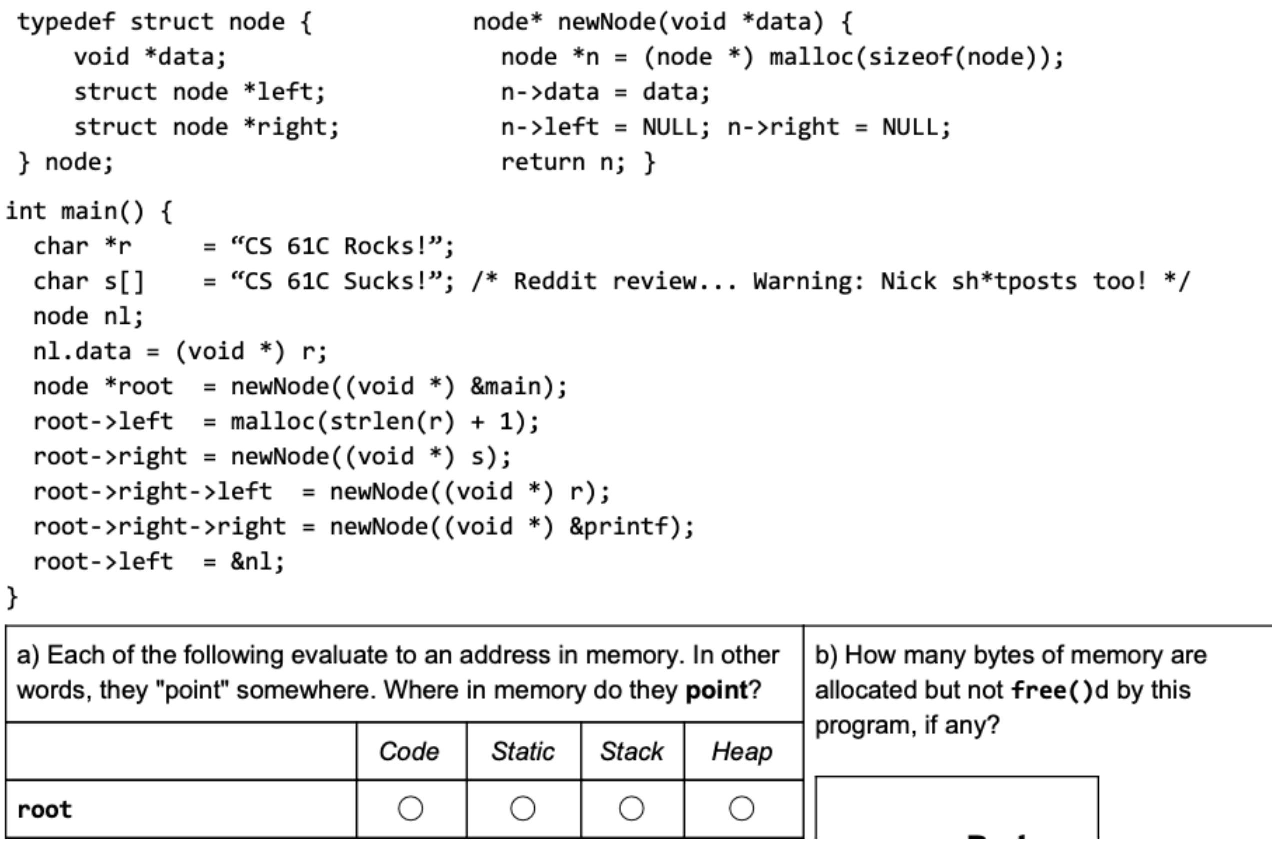
7 endorsed followup comments

● Resolved ○ Unresolved

Anonymous Atom 2 months ago

[SP-MT1]:Q4

In the solution, root is considered part of the heap.



```
typedef struct node {           node* newNode(void *data) {
    void *data;                   node *n = (node *) malloc(sizeof(node));
    struct node *left;            n->data = data;
    struct node *right;           n->left = NULL; n->right = NULL;
} node;                           return n; }

int main() {
  char *r      = “CS 61C Rocks!”;
  char s[]     = “CS 61C Sucks!”; /* Reddit review... Warning: Nick sh*tposts too! */
  node nl;
  nl.data = (void *) r;
  node *root  = newNode((void *) &main);
  root->left  = malloc(strlen(r) + 1);
  root->right = newNode((void *) s);
  root->right->left  = newNode((void *) r);
  root->right->right = newNode((void *) &printf);
  root->left  = &nl;
}
```

a) Each of the following evaluate to an address in memory. In other words, they "point" somewhere. Where in memory do they **point**?					b) How many bytes of memory are allocated but not **free()**d by this program, if any?
	Code	*Static*	*Stack*	*Heap*	
root	○	○	○	○	

However in the Homework 02 Question 4, Chekov is part of the stack and *Chekov is part of the heap.

So I don't understand why here root is part of the heap.

helpful! | 0

Anonymous Helix 2 months ago I think this question is asking about *root instead of root itself ("where in mem do they point"), so root would be stored on the stack but it is pointing to a node on the heap (created in the newNode function).

helpful! | 0

Xxxxxxx 2 months ago The words may sound confusing but, the question is asking for where root "points" to. As root holds an address to the dynamic allocated memory, it's "pointing" to the heap. But if it asks where root is "stored", the answer should be definitely the stack.

helpful! | 1

● Resolved ○ Unresolved

Anonymous Mouse 2 months ago

[SU-MT1]:Q3.2

The if statement is supposed to make ret_val evaluate to true on all false values of i and and false on on all true values of i. However, it seems to me like there should be an error. The conditional checks if i is equal to 0, which is a false value and if so, it sets ret_val to false. Why would this not be a logical error?

```
2.

/* Function that takes in an integer, interprets it as a boolean value,
 * and returns a string that can be dereferenced outside the function
 * indicating if it was true or false.*/
char* bool_to_string (int i) {
     /* Allocates space for a pointer. */
[ ]  char* ret_val; // Allocates space for a pointer but not contents

     /* Evaluates to true on all false values and false on all true values. */
[ ]  if (i == 0) {
         ret_val = "false";
     } else {
         ret_val = "true";
     }
     // String literals have memory allocated, so the assignment works

     /* Returns a pointer that can be dereferenced in other functions. */
[ ]  return ret_val; // String literals last for the life of the program
}

[ ] no errors
```

helpful! | 1

Xxxxxxx 2 months ago `ret_val` is supposed to indicate whether i, when interpreted as a bool, is "true" or "false" (comment above the function). I believe the comment you mention `/* Evaluates to true on all false values and and false on on all true values.*/` is referring to the line `if (i == 0)` in the code and not `ret_val` itself.

helpful! | 0

Anonymous Atom 2 2 months ago Isn't this very ambiguous for an exam question then? When I was doing this question, it seemed very clear to me that it's a logical error since it evaluates false on false values instead of true values, so I wouldn't even ask for a clarification.

helpful! | 0

● Resolved ○ Unresolved

Xxxxxxx 2 months ago

I tried to do something for Spring 2019 midterm 2, but I couldn't finish because I has other class. Hopefully this helps someone.

https://docs.google.com/document/d/1sgZ4-NACuqerBr-nrbfY3krotL0mnoS7dB74bS2g9rl/edit?usp=sharing

helpful! | 3

Anonymous Mouse 3 1 month ago Thank you so much!!!!!

helpful! | 0

● Resolved ○ Unresolved

Anonymous Beaker 2 months ago

Rendering markdown...

helpful! | 1

Xxxxxxx 2 months ago I think it's asking for where `map[1].value` is stored. SInce it's in the array of `map_entry` allocated on the heap, heap is the answer.
The elements in `char value[20]` are on the stack.

helpful! | 0

Anonymous Gear 2 months ago So if we did **map[0].key**, would that also be heap? And if this question asked for ***map[1].value**, would that be on the stack?

helpful! | 0

Anonymous Scale 2 months ago Correct me if I'm wrong, but it seems to be the case that we are storing the pointers (i.e. the addresses to values) on the heap when setting `m[size].key = k` and `m[size].value = v`. This means that the addresses should be stored on the heap since `m` is on the heap. I would think that `map[0].key` would be on the heap and that `*map[1].value` would be on the stack.

~ An instructor (Jenny) thinks this is a good comment ~

helpful! | 2

● Resolved ○ Unresolved

Anonymous Calc 2 months ago
Sp19,

Q6:
"

The 61Ccc compiler you wrote in projects 1-1 and 1-2 is considered a complete compiler in that it converts c-like les to assembly code.
"

False.

Does not the compiler convert c-file to assembly code?

helpful! | 0

Xxxxxxx 2 months ago It was specific for Spring 2019. You can just ignore this question. In case you are curious about it, in project 1-1 and 1-2 we only finished the parser and the tokenizer. It was not considered complete until we finished project 2, in which we generated risc-v code from complex expressions.

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

helpful! | 1

● Resolved ○ Unresolved

Anonymous Calc 2 months ago
SP19 P2 e)- a) map is in heap and zeros is in stack, but why map could be lager than zero?

helpful! | 1

Xxxxxxx 2 months ago Be careful with the words here. It's not asking for where these variables are stored. Instead, you need to consider what these values evaluate to.
`zero` holds 0 while `map` holds a value greater than 0. So `map > zero` will evaluate to true.

helpful! | 0

Anonymous Comp 2 months ago Doesn't zero hold NULL and not 0?

helpful! | 0

● Resolved ○ Unresolved

Anonymous Calc 2 months ago

I see expression "a0/s0" in sp19 P3, is "/ " the division?

helpful! | 0

> **Xxxxxxx** 2 months ago No, it means "or"
>
> ~ An instructor (Caroline Liu) thinks this is a good comment ~
>
> helpful! | 1

● Resolved ○ Unresolved

Anonymous Comp 2 months ago

[SU19]: Q4.2

For the second line, why do we reference text with a period instead of an arrow? I know array[i] refers to the caption_t struct, but isn't text a pointer stored within that struct?

helpful! | 0

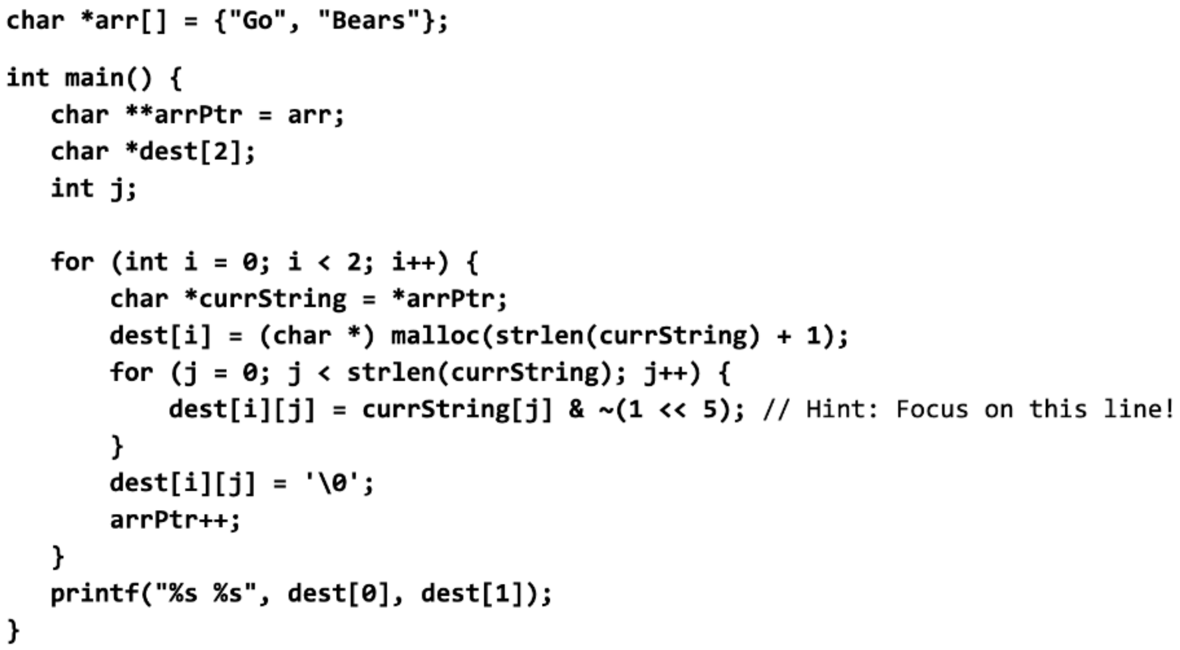
> **Caroline Liu** 2 months ago You can access items within a struct in multiple ways. Given struct A with fields B and C, and A isn't a pointer, you can say A.B or A.C. However, if we have struct_name *A, we can access B and C as such: A->B and A->C. HOWEVER, the arrow operator is shorthand for (*A).B and (*A).C. Hope that makes more sense!
>
> helpful! | 0

● Resolved ○ Unresolved

Anonymous Poet 2 months ago

[FA19]: Q5 (quest)

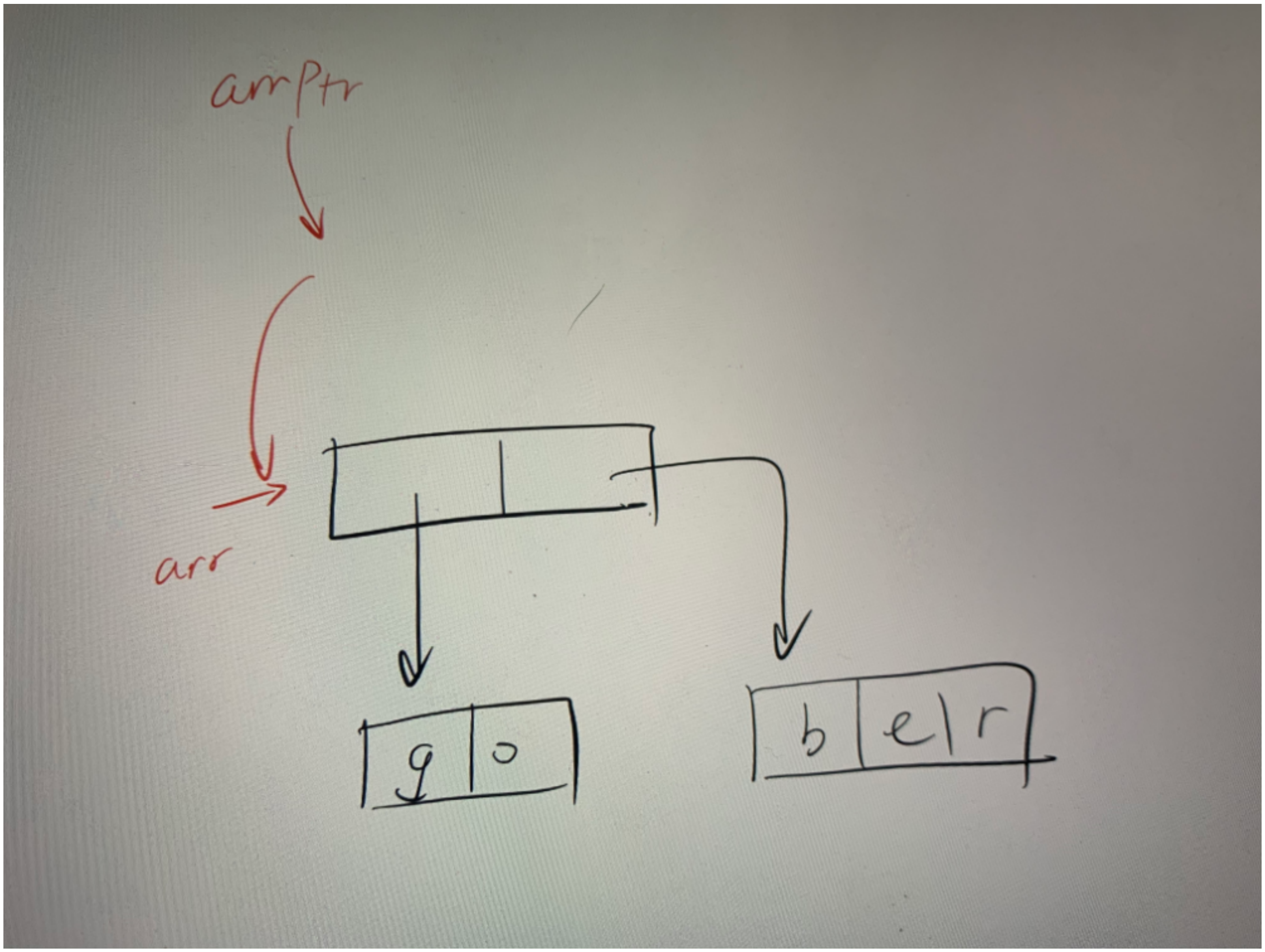
At the very end of outer for loop, why would attrPointer++ move the pointer to the next element in arr? If **arrPtr = arr sets arrPtr to point to another pointer P that points to the array, doesn't attPtr++ advance P but not the one that points to the array? I drew out the diagram and it's probably wrong lol. Thank you!



```
char *arr[] = {"Go", "Bears"};

int main() {
   char **arrPtr = arr;
   char *dest[2];
   int j;

   for (int i = 0; i < 2; i++) {
       char *currString = *arrPtr;
       dest[i] = (char *) malloc(strlen(currString) + 1);
       for (j = 0; j < strlen(currString); j++) {
           dest[i][j] = currString[j] & ~(1 << 5); // Hint: Focus on this line!
       }
       dest[i][j] = '\0';
       arrPtr++;
   }
   printf("%s %s", dest[0], dest[1]);
}
```



helpful! | 0

Anonymous Atom 2 2 months ago char** arrptr = arr; sets arrptr to point to the same array that arr is pointing to. Thus, currString is "Go" (when the ptr isn't moved yet) and currString[0] is "G". To accomplish your diagram, you want: char** arrptr = &arr.

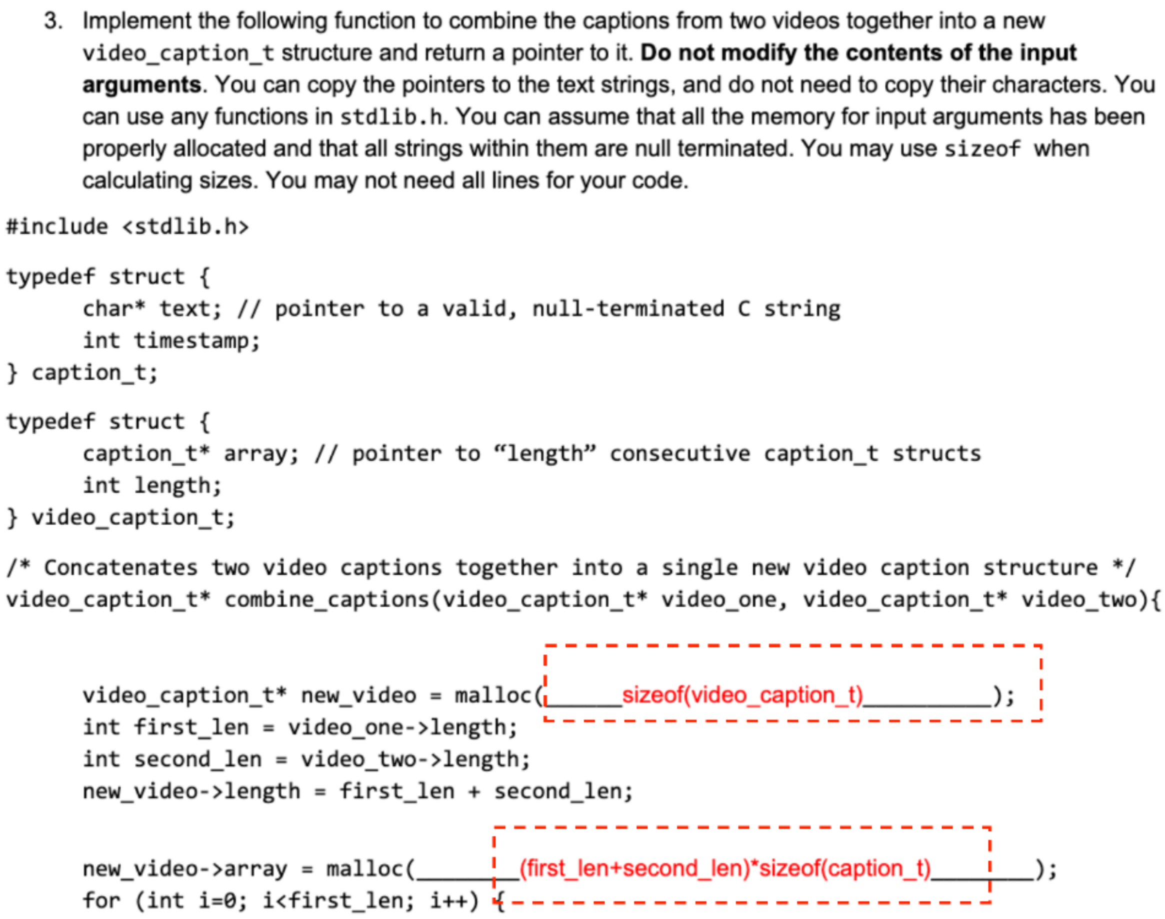
helpful! | 0

● Resolved ○ Unresolved

Anonymous Helix 2 2 months ago

For this question, why do we only need to malloc one unit of video_caption_t, but first_len + second_len units of caption_t?

Thanks!



3. Implement the following function to combine the captions from two videos together into a new `video_caption_t` structure and return a pointer to it. **Do not modify the contents of the input arguments**. You can copy the pointers to the text strings, and do not need to copy their characters. You can use any functions in `stdlib.h`. You can assume that all the memory for input arguments has been properly allocated and that all strings within them are null terminated. You may use `sizeof` when calculating sizes. You may not need all lines for your code.

```
#include <stdlib.h>

typedef struct {
      char* text; // pointer to a valid, null-terminated C string
      int timestamp;
} caption_t;

typedef struct {
      caption_t* array; // pointer to “length” consecutive caption_t structs
      int length;
} video_caption_t;

/* Concatenates two video captions together into a single new video caption structure */
video_caption_t* combine_captions(video_caption_t* video_one, video_caption_t* video_two){


      video_caption_t* new_video = malloc(______sizeof(video_caption_t)__________);
      int first_len = video_one->length;
      int second_len = video_two->length;
      new_video->length = first_len + second_len;

      new_video->array = malloc(________(first_len+second_len)*sizeof(caption_t)________);
      for (int i=0; i<first_len; i++) {
```

helpful! | 0

Xxxxxxx 2 months ago **[SU-MT1]:Q3**
You only need to malloc one unit of video_caption_t, since you are only allocating the space needed to create a single new video caption structure.

The goal of this question is to concatenate two video captions together. For our new caption_t array, we expect it to hold all of the caption_t elements in both the first and second video captions. That's why we need to allocate `(first_len + second_len) * sizeof(caption_t)` space.

helpful! | 0

● Resolved ○ Unresolved

Anonymous Mouse 2 2 months ago
[Su19]: Q2.1
What's the difference between &is_complete and is_complete when talking about the corresponding memory region?

helpful! | 1

Jenny 2 months ago is_complete is a boolean variable which has value true or false. This question asks "State in which region of memory each value corresponds to", because is_complete does not evaluate to an address value, there is really no point of asking this question. However, &is_complete evaluates to the address of is_complete, which shall be static.

helpful! | 0

● Resolved ○ Unresolved

Anonymous Beaker 2 2 months ago

For SU19 3.3 why is the line copy[i] = *(str++) correct? Isn't it incrementing before dereferencing when we want to dereference first then increment the string??

helpful! | 0

Anonymous Calc 2 2 months ago Have the same question. I thought that should be false.

helpful! | 0

Anonymous Beaker 2 2 months ago @ Anonymous calc

It should actually be right! Just checked prefix/postfix slides. We can think of *(str++) as:

copy[i] = *str
str++

~ An instructor (Jenny) thinks this is a good comment ~

helpful! | 1

● Resolved ○ Unresolved

Anonymous Beaker 2 2 months ago
Does anyone know why the double pointer on 4.1 SU19 has size of 4? I thought double pointers had size of 8

helpful! | 0

Anonymous Comp 2 2 months ago all pointers have size 4. double pointers, which are pointers to pointers, are no exception.

~ An instructor (Jenny) thinks this is a good comment ~

helpful! | 2

● Resolved ○ Unresolved

Anonymous Calc 2 2 months ago
[SU-MT1]:Q5

Why does the End label should do jr ra? I thought it shouldn't matter?

helpful! | 0

● Resolved ○ Unresolved

Anonymous Comp 1 month ago
[FA-MT2]: Q4

Instead of doing

```
mv a0, t0
```

could we have done

```
addi a0, a0, 4
```

?

helpful! | 0

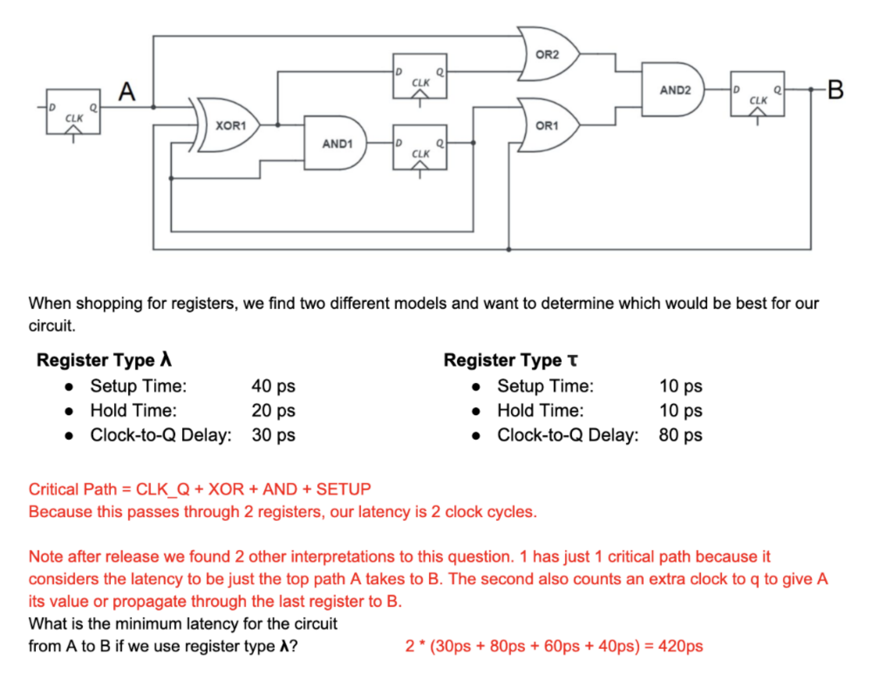
● Resolved ○ Unresolved

Anonymous Gear 1 month ago

[SU-MT2]: Q3

helpful! | 0

Anonymous Gear 1 month ago



When shopping for registers, we find two different models and want to determine which would be best for our circuit.

Register Type λ

Setup Time:	40 ps
Hold Time:	20 ps
Clock-to-Q Delay:	30 ps

Register Type τ

Setup Time:	10 ps
Hold Time:	10 ps
Clock-to-Q Delay:	80 ps

Critical Path = CLK_Q + XOR + AND + SETUP
Because this passes through 2 registers, our latency is 2 clock cycles.

Note after release we found 2 other interpretations to this question. 1 has just 1 critical path because it considers the latency to be just the top path A takes to B. The second also counts an extra clock to q to give A its value or propagate through the last register to B.

What is the minimum latency for the circuit from A to B if we use register type λ? 2 * (30ps + 80ps + 60ps + 40ps) = 420ps

"Because this passes through 2 registers, our latency is 2 clock cycles"
What are the 2 registers that this is referring to? Is it the one after AND1 and the one after AND2? Also just to make sure I understand the solution, it's basically just saying that latency is the number of clock cycles it takes to go from A to B, and to calculate the minimum latency in ps, we just need to multiply that number of clock cycles * the minimum clock time (clock to q + longest CL + setup time)?

And if this was the case, why were the other interpretations they mention in the question valid? The top path is not the longest CL, therefore it shouldn't be seen as the critical path. Also, why would adding in an extra clock to q be valid? Wouldn't that already be accounted for when we calculated the critical path?

Thanks!

helpful! | 1

● Resolved ○ Unresolved

Anonymous Gear 2 1 month ago

SU_MT2: For calculating the latency, we need to find out the critical path. For the first calculation of latency using critical path, my answer is different from the answer given in the sheet. For the register type `lambda`: critical path = 30 + 80 + 60 + 40 + 30 + 40 + 60 + 40 + 30 = 410.

helpful! | 0

Anonymous Gear 1 month ago I think the critical path is just the longest path between *registers* rather than the longest path between A and B. Therefore, the longest path between registers should be from the initial register at A to the bottom middle register (after the AND1 gate), which would be clq-to-q delay + XOR delay + AND delay + setup time = 210 ps.

helpful! | 4

Anonymous Helix 4 1 month ago I have 210 too, but the answer gave us is 420. I am confused why need to multiply 2

helpful! | 0

● Resolved ○ Unresolved

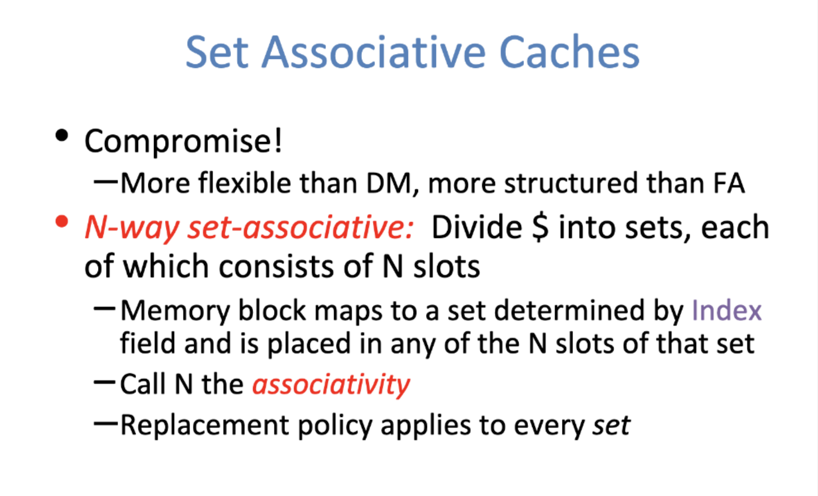
Anonymous Mouse 2 1 month ago

SU_MT2 Q5.5:

When we say the number of associativity, does this mean the number of slots of one set or number of sets? I'm also confused about the solution saying "Notice that 0, 1, 2, 3 never conflict mod 4, so each block of arr must be in a unique set." Since we only have two sets, how can each block of arr be in a unique set? Why 0, 1, 2, 3 never conflict mod 4?

helpful! | 0

Anonymous Gear 1 month ago Number of associativity = number of slots in one set. From lecture 16:



Set Associative Caches

- Compromise!
  - —More flexible than DM, more structured than FA
- *N-way set-associative:* Divide $ into sets, each of which consists of N slots
  - —Memory block maps to a set determined by Index field and is placed in any of the N slots of that set
  - —Call N the *associativity*
  - —Replacement policy applies to every *set*

Not sure how to answer your second question since I'm a bit confused about the solution as well :(

helpful! | 0

Xxxxxxx 1 month ago The solution confused me too, here is how I approach this:

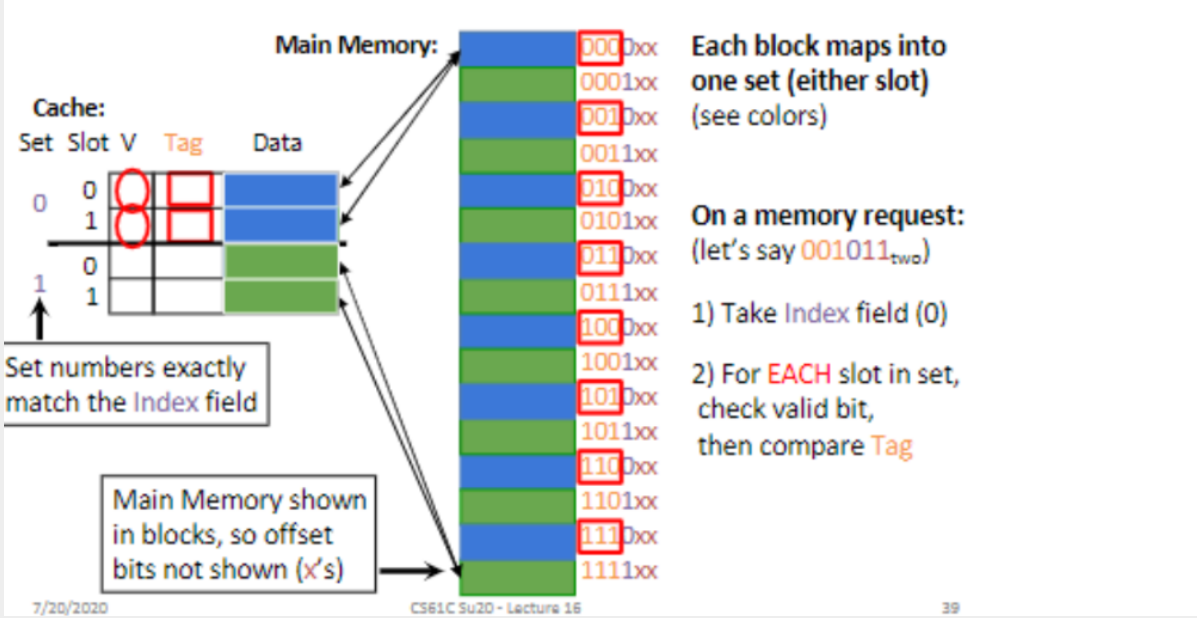
The Direct map is not the case here because in worst case temp and arr may get into the same slot and cause conflict. But if there are 4 sets with 2 slots in each, there are only compulsory misses, and the hit rate stays the same.

helpful! | 2

Anonymous Helix 3 1 month ago How do you know that each block of arr maps to a different set?

helpful! | 0

Xxxxxxx 1 month ago



Each block maps into one set (either slot) (see colors)

On a memory request: (let's say 001011_{two})

1) Take Index field (0)

2) For EACH slot in set, check valid bit, then compare Tag

Set numbers exactly match the Index field

Main Memory shown in blocks, so offset bits not shown (x's)

7/20/2020 CS61C Su20 - Lecture 16 39

The alignment of the array makes it so. For the exam problem, say the address of the first element is 0...000000(first block), then the address of the fifth element should be 0...010000(second block). The remaining blocks follow the same pattern (index bits will update when one block ends). So each block maps to different set,

helpful! | 1

● Resolved ○ Unresolved

Anonymous Scale 2 1 month ago

SU_MT2 Q3.4

Why the Out_2 delay was 260ps? I did simplification but I can't get the correct answer...

helpful! | 0

Anonymous Gear 1 month ago For the Out_2 delay, I found the longest combinatorial logic path, which goes through XOR1 -> AND1 -> OR1 -> XOR4, which would be 80 + 60 + 40 + 80 = 260 ps.

helpful! | 2

Anonymous Calc 3 1 month ago Why wouldn't we take into account XOR3? Is it because the AND1 + OR1 take longer than XOR3? And those steps happen in parallel?

helpful! | 0

● Resolved ○ Unresolved

Anonymous Poet 2 1 month ago

[SP18-MT2-Problem 3] Why is jalr correct? It usually takes in the register and immediate, so signals 0, 1. For the reversed signals 1, 0, jalr would be taking in PC and rs2. I am wondering why that is not invalid?

helpful! | 0

Anonymous Atom 3 1 month ago +1

helpful! | 0

Anonymous Mouse 3 1 month ago +1

helpful! | 0

● Resolved ○ Unresolved

Anonymous Helix 3 1 month ago [SU-MT2]:Q5.4
Why is it only compulsory misses? For the worst case in an N-way set associative cache, isn't there a repeating pattern of at least N+1 that maps into same set, meaning we will have to replace some blocks as we are working with six blocks and there are four blocks in a single set?
helpful! | 1

> **Anonymous Comp 3** 1 month ago I'm guessing that since the index bit is just one bit and the array is continuous, then the index for blocks 0 1 2 3 will be 0 1 0 1 because their addresses will be xxxx00xxxx xxxx01xxxx xxxx10xxxx xxxx11xxxx where we only consider one bit as the index bit. So every set will have 2 blocks occupied by the array.
> helpful! | 0

● Resolved ○ Unresolved

Anonymous Beaker 2 1 month ago
[SU-MT2]:Q3 Should we not bother simplifying the boolean expression when trying to find the correct circuit for similar problems? Because I initially tried to simplify at first but it got me farther away from the correct answer and doing it without simplifying got me to the correct answer.
helpful! | 0

● Resolved ○ Unresolved

Anonymous Beaker 3 1 month ago
[SP-MT2]: Q 1.3
I'm having trouble reaching the solution posted. My boolean algebra simplification goes like ~(~(~(A * B) + C) * D) = ~(~(~A + ~B + C) * D) = ~(A * B * ~C * D) = ~A + ~B + C + ~D, which is answer A, however, the solutions say the correct answer is C. Why is this the case?
helpful! | 0

> **Xxxxxxx** 1 month ago C is equivalent to A and it's simplest, It only takes 3 gates, comapared to A using 6 gates.
> helpful! | 0

● Resolved ○ Unresolved

Anonymous Poet 1 month ago
[SU-MT2 Q5.6]
I'm having trouble understanding global miss rate. The denominator of L2 global miss rate is # accesses to the cache system. Is it equal to (# accesses to L1) or (# accesses to L1 + # access to L2)? The TAs brought up the idea of "double counting" in the review session. If we use double counting in this problem, # accesses to L1 should be 200 instead of 300 and L1 miss rate should be 1/2 instead of 1/3.
helpful! | 0

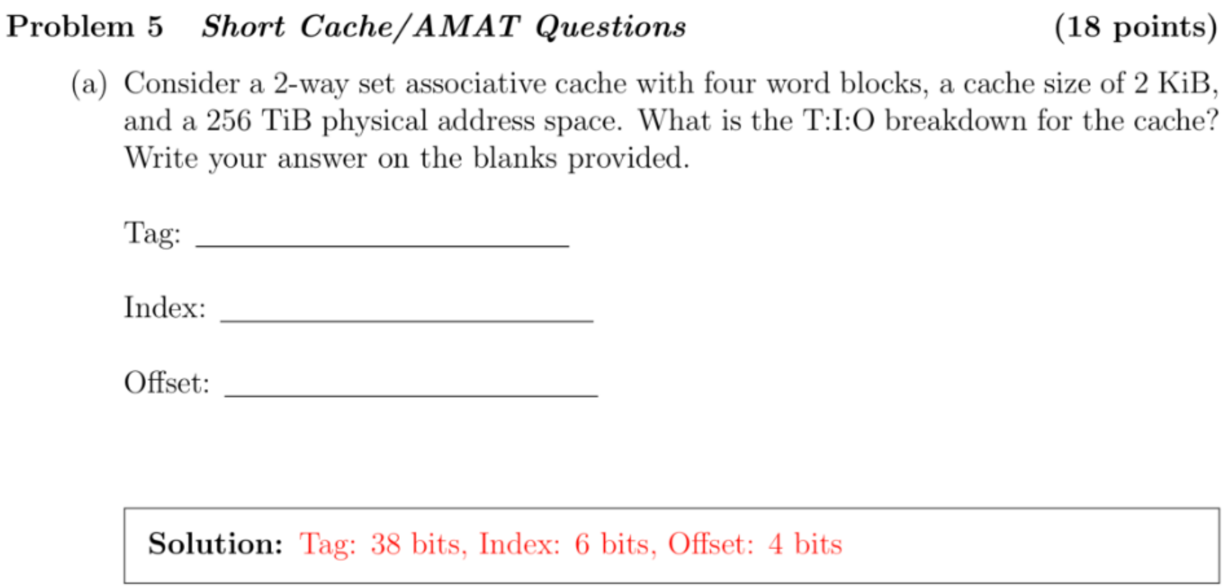
● Resolved ○ Unresolved

Anonymous Beaker 2 1 month ago
For SU19 MT2 question 5.3 I'm confused about the worse case HR. In the solution it says there is no limit and our HR will be maintained no matter the size but when I calculated each of them individually the HR increased. So why isn't the answer size = 16 then?
helpful! | 0

● Resolved ○ Unresolved

Anonymous Atom 2 1 month ago

[SP-MT2]



Problem 5 ***Short Cache/AMAT Questions*** **(18 points)**

(a) Consider a 2-way set associative cache with four word blocks, a cache size of 2 KiB, and a 256 TiB physical address space. What is the T:I:O breakdown for the cache? Write your answer on the blanks provided.

Tag: ______________________

Index: ______________________

Offset: ______________________

Solution: Tag: 38 bits, Index: 6 bits, Offset: 4 bits

Why is the offset bits 4 instead of 5?

helpful! | 0

Anonymous Scale 1 month ago 4 words/block = 4 bytes/word * 4 words/block = 16 bytes/block

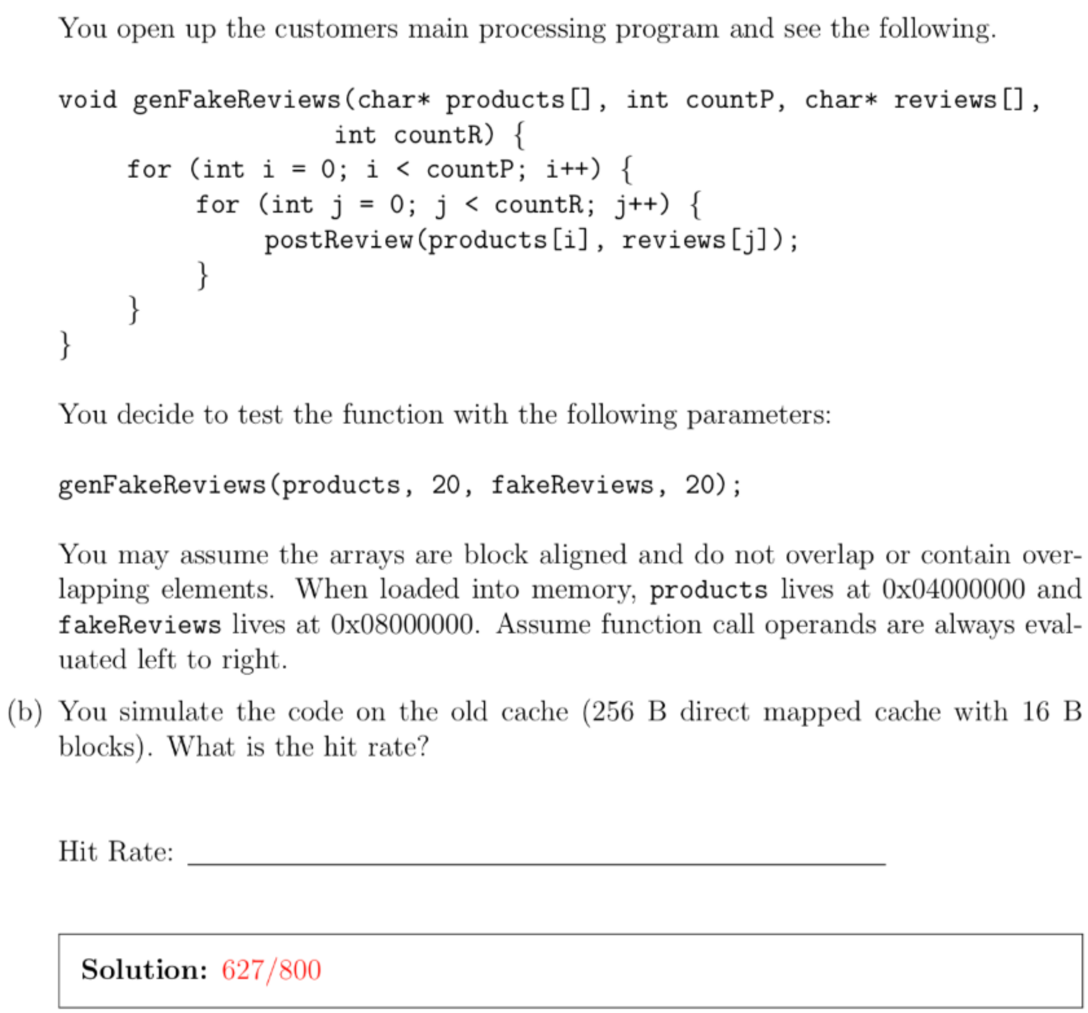
log_2(16) = 4 offset bits

helpful! | 1

● Resolved ○ Unresolved

Anonymous Atom 2 1 month ago

[[SP-MT2]



You open up the customers main processing program and see the following.

```
void genFakeReviews(char* products[], int countP, char* reviews[],
                    int countR) {
     for (int i = 0; i < countP; i++) {
          for (int j = 0; j < countR; j++) {
               postReview(products[i], reviews[j]);
          }
     }
}
```

You decide to test the function with the following parameters:

```
genFakeReviews(products, 20, fakeReviews, 20);
```

You may assume the arrays are block aligned and do not overlap or contain overlapping elements. When loaded into memory, **products** lives at 0x04000000 and **fakeReviews** lives at 0x08000000. Assume function call operands are always evaluated left to right.

(b) You simulate the code on the old cache (256 B direct mapped cache with 16 B blocks). What is the hit rate?

Hit Rate: ______________________________

Solution: 627/800

Can someone explain how to get to this answer? In my calculations, the hit rate is quite low since the same block gets replaced almost every time.

helpful! | 0

Anonymous Scale 1 month ago Note that this is an array of char* pointers, so each entry is 4 bytes rather than 1 byte. See if this changes your calculations.

helpful! | 0

Anonymous Atom 2 1 month ago That changes something! However, I'm still not getting the correct answer. I got that when i=0, there're 13 misses, and for each i that follows, there're 10 misses each round, which gives 13+19*10 = 223 misses out of the 800 access, still not equal to the correct 173 misses. Is this question that complicated?

helpful! | 0

Anonymous Scale 1 month ago I didn't get the exact answer but I got really really close (which I believe still gives you points for this specific question). I got 166 misses because I got 14 initial misses for i = 0, j = 0, 1, 2, 3 and from then on I got 8 misses for every value of i (so 19 values). I couldn't find a nice pattern that led me to the right answer, and I kind of put it aside since it seemed like a lot of messy casework to get to the right solution (and plus I got close enough).

helpful! | 0

Anonymous Poet 3 1 month ago When i = 0, you have 13 misses for pairs of (i, j): (0,0), (0,1), (0,2), (0,3), (0,4) and for j = 8, 12, 16.

Afterward, for every i mod 4 $\neq$ 0, you have 8 misses. For example when i = 1, you miss j = 0 for (1,0), pairs (1,1), (1,2), (1,3), and i = 4 for (1,4).

For every i mod 4 = 0, you have 10 misses. For example when i = 4, you miss (4,0), j =4 for (4,4) , (4,5), (4,6), (4,7), i = 4 for (4,8).

In total, you have 13 + 8 x 15 + 10 x 4 = 173 misses.

~ An instructor (Daniel Fan) thinks this is a good comment ~

helpful! | 2

● Resolved ○ Unresolved

Anonymous Beaker 1 month ago

Rendering markdown...

helpful! | 0

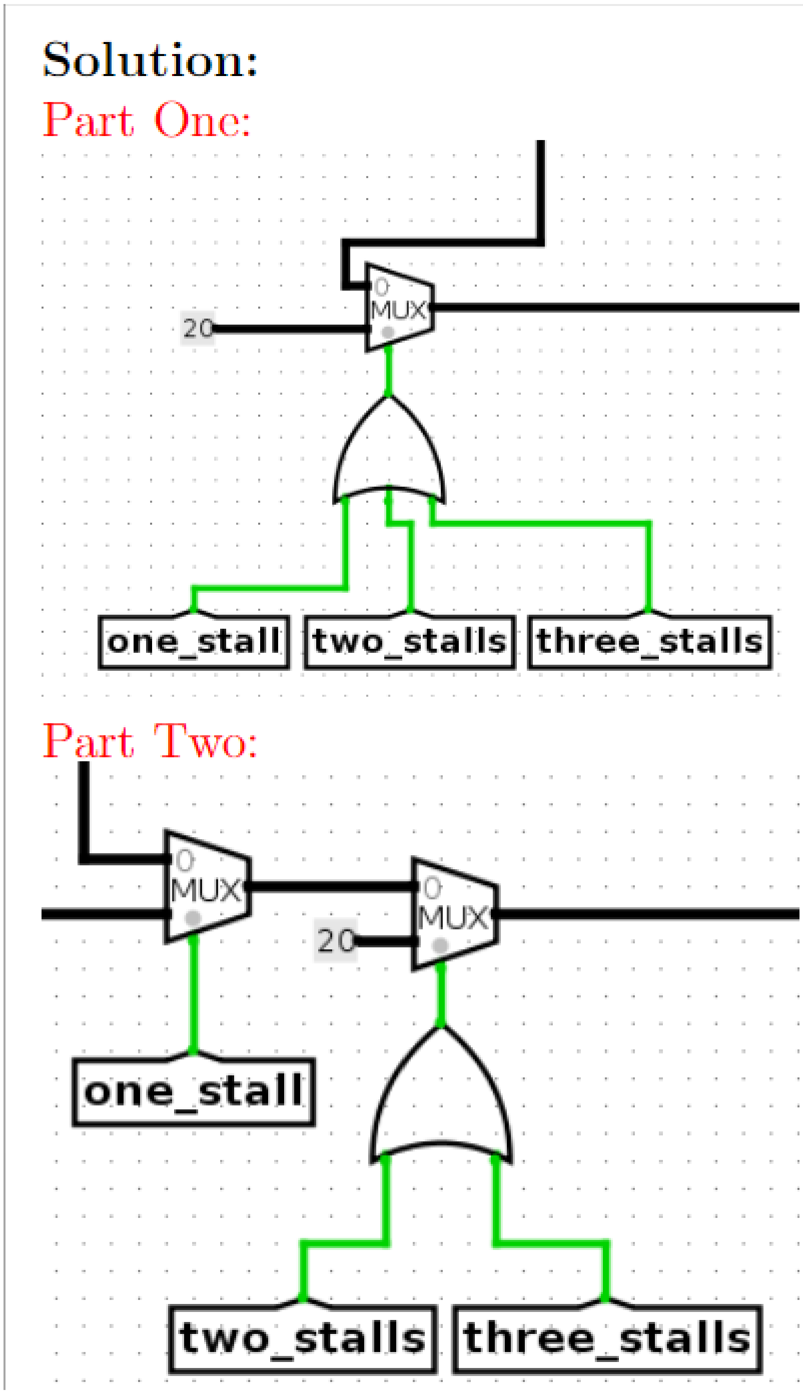
Xxxxxxx 1 month ago 1 is the zeroth power of 4.

helpful! | 0

● Resolved ○ Unresolved

Anonymous Gear 3 1 month ago

[SP-MT2] Q7.



Solution:

Part One:

Part Two:

Can someone explain how this solution works?

helpful! | 0

● Resolved ○ Unresolved

Anonymous Scale 3 1 month ago

[SU-MT2] Q5.2

Why isn't the number of compulsory misses 8 instead of 4? I thought in the worst case hit rate, we are trying to get as much compulsory misses as possible.

helpful! | 0

Xxxxxxx 1 month ago Each block is 16 bytes, and an array of 16 ints is only 64 bytes, so we'll only have to insert 4 blocks into the cache.

Also, according to the problem description, "Assume that all loads are executed from left to right, **for all questions any arrays are block aligned**, and that sizeof(int) == 4." This means no funny business with how the ints are contained in blocks; the first four ints are in the first block we bring in, the last 4 are in the second (i.e. from the access of arr[size-i-1]), etc.

helpful! | 1

● Resolved ○ Unresolved

Anonymous Atom 4 1 month ago

[FA-F] Q8b: Why is there a conflict miss when we read ARRAY[i+256]? My understanding is that the address of ARRAY[i+256] will be at a new index since the offset size is only 4 bits, giving us a compulsory miss on the first iteration, but in future iterations we would have already loaded ARRAY[i] into cache since we're incrementing by i += 256 per iteration

helpful! | 0

Xxxxxxx 1 month ago ARRAY[i+256] will get into same slot as ARRAY[i].

helpful! | 0

Xxxxxxx 28 days ago Think about the number of bytes that 256 indices away will take and how big the cache is. 256 = 2^8 and each integer is 2^2 bytes. Once we get the number of bytes away i + 256 is from i, we've reached the size of our cache. This will bring us back to the index that array[i] is in.

helpful! | 0

● Resolved ○ Unresolved

Anonymous Mouse 4 1 month ago

[FA-F] A lot of the questions on the 2019 Summer Final seem to reference a handout for its problems. Does anyone know where to find the handout, or if it hasn't been released, then could instructors look into making that available so we can work through the problems?

helpful! | 0

Justin Cheng 1 month ago Your question is tagged [FA-F], but you mention the summer final in your post. Which final are you referring to?

helpful! | 0

Sunay Poole 1 month ago Added links to Spring's and Summer's in the post.

helpful! | 0

● Resolved ○ Unresolved

This was marked a duplicate to the question/note above by Jerry Xu 29 days ago

""

Anonymous Beaker 4 29 days ago "handout" on su19 final

Rendering markdown...

helpful! | 0

Jerry Xu 29 days ago @433

helpful! | 0

● Resolved ○ Unresolved

Xxxxxxx 29 days ago
[SU19-Final]:Q1.8:

I thought in order for a number to be in the range[0, 1), it has to be a denorm? So doesn't the exponent have to be 0?

helpful! | 0

Sunay Poole 28 days ago Nope! Remember that our true exponent = exp - bias (where exp is the thing that gets stored in FP format). So, if we just have a negative true exponent, we'll be able to represent numbers in the [0, 1) range. I recommend watching the video walkthrough linked above for this one if this is still confusing - I step through an explanation for each answer.

helpful! | 0

Anonymous Calc 4 28 days ago I watched the video, and now I just wanna confirm - denorms don't cover ALL the decimal numbers between 0 and 1?

helpful! | 0

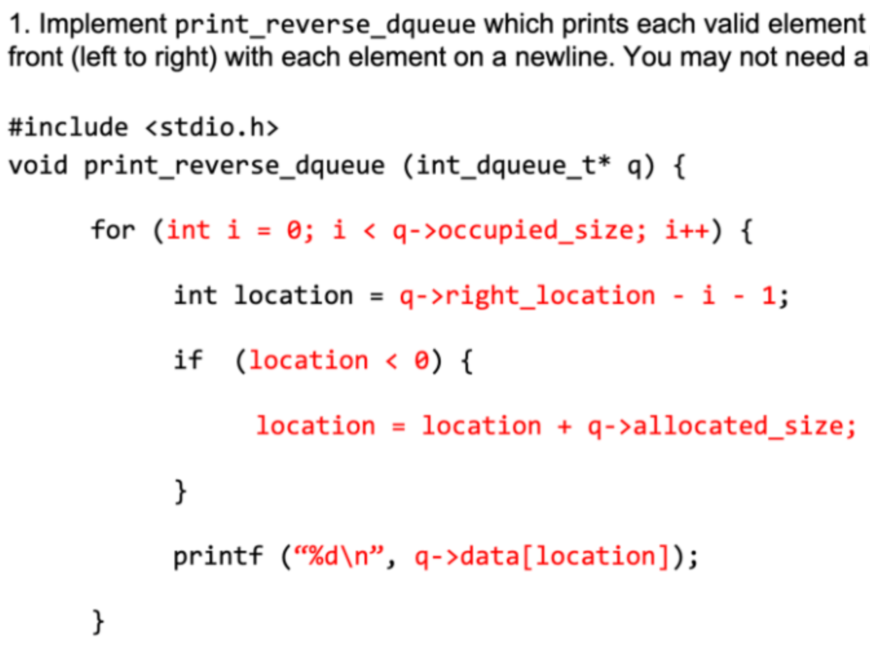
Sunay Poole 28 days ago Nope! They're only used to represent numbers one order smaller than what we can with normalized numbers. Since denorms have such a small exponent, they actually can't represent numbers far from 0.

Notice that even with the implicit 1 with normalized numbers, since our true exponent ranges from -127 to 128, we are still able to represent numbers smaller than 1. i.e. If we want to represent 0.25, this would be $(-1)^0 * 1.00000... * 2^{-2}$.

helpful! | 0

● Resolved ○ Unresolved

Xxxxxxx 29 days ago
[SU19-Final]:Q3
Isn't starting from "location = q->right_location - i - 1" printing from the right? The problem said to print from "left to right".



1. Implement `print_reverse_dqueue` which prints each valid element in the array from the end to the front (left to right) with each element on a newline. You may not need all lines.

```
#include <stdio.h>
void print_reverse_dqueue (int_dqueue_t* q) {

     for (int i = 0; i < q->occupied_size; i++) {

          int location = q->right_location - i - 1;

          if  (location < 0) {

               location = location + q->allocated_size;

          }

          printf (“%d\n”, q->data[location]);

     }
```

helpful! | 0

Sunay Poole 28 days ago Hm, not too sure what the question meant by left to right, but the idea is we want to print the dqueue in reverse. Recall that the q->right_location represents the next location at the end where we'd add an element. So, in order to print in reverse, we'd want to start right before the right_location and proceed toward the front.

helpful! | 0

● Resolved ○ Unresolved

Xxxxxxx 28 days ago

[Sp 19 Final]: Q8 d

Why is it 2 way set associative? I thought it would be 4 way set associative since having 4 blocks per set would allow you to store the respective blocks for arr, A, B, and C without eviction. Please clarify.

helpful! | 0

Daniel Fan 28 days ago I covered this in the video walkthrough, but the question asks about the "best case hit rate" and there exists a sequence of accesses where 2 way set associativity is sufficient.

helpful! | 1

Xxxxxxx 28 days ago Makes sense. Just watched the walkthrough, thank you Daniel.

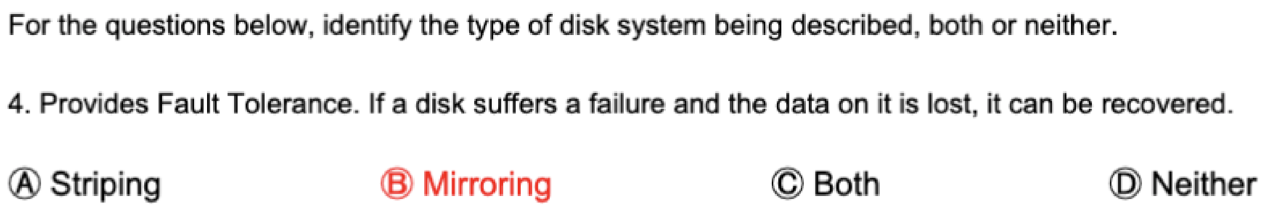
helpful! | 0

● Resolved ○ Unresolved

Anonymous Calc 4 28 days ago

[SU19-Final]Q6.4

If you have striping with parity bits, doesn't that help you recover a lost bit?



For the questions below, identify the type of disk system being described, both or neither.

4. Provides Fault Tolerance. If a disk suffers a failure and the data on it is lost, it can be recovered.

Ⓐ Striping Ⓑ Mirroring Ⓒ Both Ⓓ Neither

helpful! | 0

Xxxxxxx 28 days ago Question probably means striping alone, i.e RAID 0.

~ An instructor (Sunay Poole) thinks this is a good comment ~

helpful! | 1

Anonymous Calc 4 28 days ago o

helpful! | 0

● Resolved ○ Unresolved

Xxxxxxx 28 days ago

[Fall 19 Final] Q 4c / d

I don't understand how a0 eventually becomes 0. Doesn't -(2^32) - 1 overflow to 2^32 - 1 and not 0 if you keep subtracting?

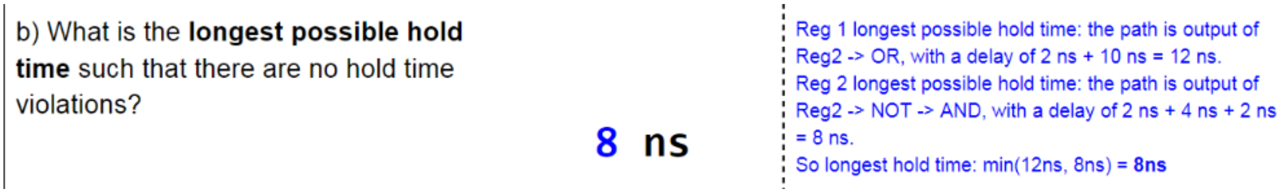
helpful! | 0

Xxxxxxx 28 days ago It goes from -1 to -2^31, then from 2^31 - 1 to 0. Therefore there are 2^32 calls, plus the first 10 calls there are 2^32 + 10 in total.
The solution says -1 or 2^32 - 1 really means if you interpret the value at a0 as an unsigned number, it will be 2^32 - 1 (all 1s).
helpful! | 1

Xxxxxxx 28 days ago Ah makes sense, thank you!
helpful! | 0

● Resolved ○ Unresolved

Anonymous Helix 28 days ago
[FA-F] Q.5b

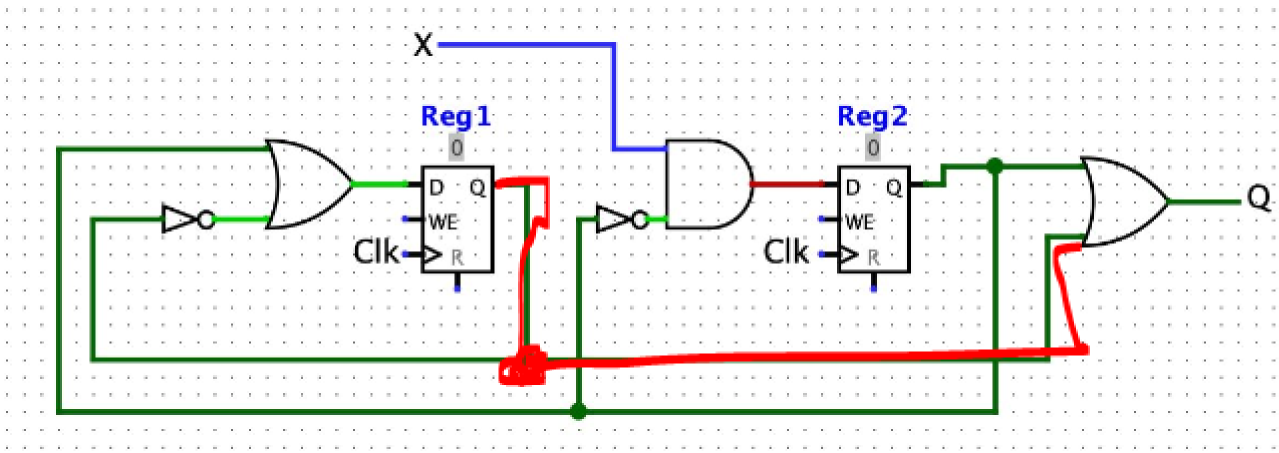


b) What is the **longest possible hold time** such that there are no hold time violations?

8 ns

Reg 1 longest possible hold time: the path is output of Reg2 -> OR, with a delay of 2 ns + 10 ns = 12 ns.
Reg 2 longest possible hold time: the path is output of Reg2 -> NOT -> AND, with a delay of 2 ns + 4 ns + 2 ns = 8 ns.
So longest hold time: min(12ns, 8ns) = **8ns**

I think the solution is referring to this red line for Reg1 longest Hold Time



Are we assuming that Q is a register that also has to stay stable for writing?
helpful! | 0

Xxxxxxx 28 days ago you're looking at the wrong OR gate. The full path is Reg2 -> OR -> Reg1 (i.e the OR gate on the left) when considering the shortest CL path going into Reg1.
helpful! | 1

Anonymous Helix 27 days ago Thanks!
helpful! | 0

● Resolved ○ Unresolved

Xxxxxxx 28 days ago
[FA-F]:Q9c

Why wouldn't we need to clear the cache valid bits? Don't we need to invalidate the cache when we switch processes in order to avoid security issues?

helpful! | 0

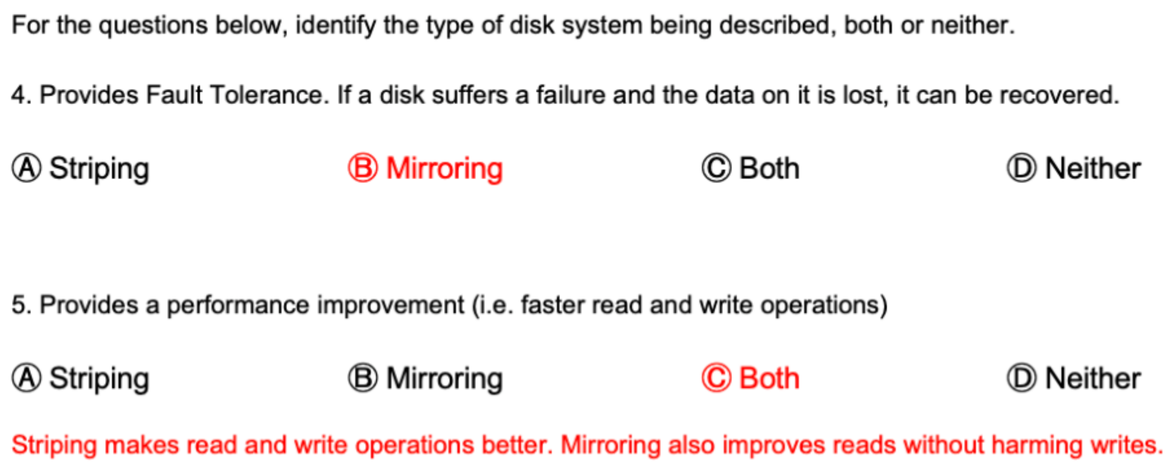
Anonymous Scale 27 days ago If switching processes you never have to clear cache. It's not a security issue because a program can't access a physical memory address, only a virtual memory address. Remember that each process has its own page table but there is only one TLB. We have to invalidate the TLB because we don't want a program accessing other programs' physical memory locations. In preventing this, we have eliminated the need to clear the cache. In other words, if a program doesn't even know that an address exists in another program, then there is no way it can try and access it in the cache.

helpful! | 0

● Resolved ○ Unresolved

Xxxxxxx 27 days ago [SU:F]:Q6.5

I am not sure why RAID 0 increases read improvement - my understanding is that you can read from either disk but the speed won't increase.



For the questions below, identify the type of disk system being described, both or neither.

4. Provides Fault Tolerance. If a disk suffers a failure and the data on it is lost, it can be recovered.

Ⓐ Striping [✓ Ⓑ Mirroring] Ⓒ Both Ⓓ Neither

5. Provides a performance improvement (i.e. faster read and write operations)

Ⓐ Striping Ⓑ Mirroring [✓ Ⓒ Both] Ⓓ Neither

Striping makes read and write operations better. Mirroring also improves reads without harming writes.

helpful! | 0

Sunay Poole 27 days ago Does 9:30 in the video walkthrough linked at the top help? Essentially, we can perform reads in parallel, giving us faster access times.

helpful! | 1

● Resolved ○ Unresolved

Anonymous Comp 27 days ago

[Sp19 Final] Q3D/E: I get Q3C and the accesses (14 for array B and 2 for array A) but I was wondering what "page table pages" meant here, and also how to solve 3E.

Thanks!

helpful! | 0

Anonymous Scale 27 days ago I'm not sure how to do 3D but here is the explanation to 3E:
You have a single level page table, so every PTE in this page table is instantiated. There are 2^VPN = 2^15 PTEs in this table. Each PTE is 4 bytes, so the total size of the page table is 2^15 PTE/table *

2^2 bytes/PTE = 2^17 bytes/table. The number of bytes per page is 128 = 2^7. Therefore the number of pages per table is 2^17/2^7 = 2^10

helpful! | 0

Anonymous Scale 27 days ago I figured out 3D: basically you're counting the number of pages that contain relevant PTEs for arrays A and B. Since the virtual addresses of A and B are so far apart, they cannot be in the same page, so their PTEs will reside in two different pages which means that you'd have to access two different pages to translate the addresses from 0x20000 and 0x30000 to whatever their physical addresses are

helpful! | 0

● Resolved ○ Unresolved

Anonymous Beaker 4 27 days ago

Rendering markdown...

helpful! | 0

Anonymous Beaker 4 27 days ago

Rendering markdown...

helpful! | 0

Max Litster 27 days ago As we saw in lab, we can drastically improve cache performance by working with the cache factors like the block size and exploiting spatial locality. A best case hit rate comes from a situation where we miss the least. Similarly, a worst case hit rate would cause our cache to miss the most.

Your second question is correct. Think about the size of our block size and the size of our array. How are they related. What ends up in the cache after the first compulsory miss?

To tie the two questions together: after our first compulsory miss, is there any situation where we will miss again?

helpful! | 0

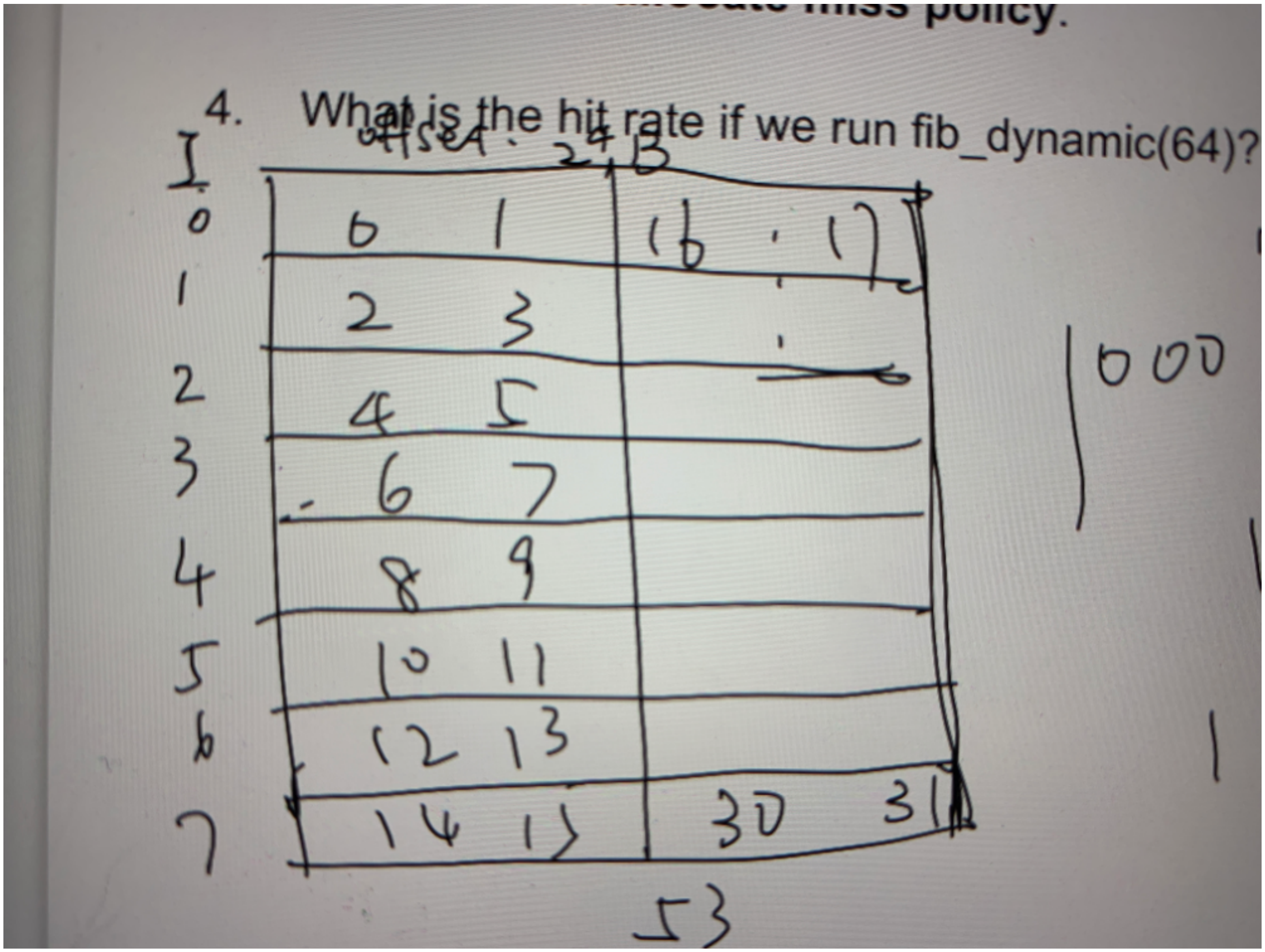
● Resolved ○ Unresolved

Anonymous Helix 4 27 days ago

[SU19-Final]Q7.5,7.6:

I have watch the walkthrough video, but still feel confused why it is N/A.

I draw a rough diagram based on my understanding. The most left is index bit, which in total are 8 entries. Since each block can only has two long numbers, f[0] and f[1] stores in the first block with index 1, then f[2] and f[3] stores in the second block with index 2, so on so forth. Then f[16] stores in the second set in the index 1 and so on. But f[32] needs to erase f[0] and f[1] since it is full. But why there will no any capacity and conflict miss but only compulsory misses? Thank you~



miss policy.

4. What is the hit rate if we run fib_dynamic(64)?

offset : 2, 4B

I		
0	0 1	16 : 17
1	2 3	:
2	4 5	
3	6 7	
4	8 9	
5	10 11	
6	12 13	
7	14 15	30 31

53

000

helpful! | 0