You can find the past exams here: https://cs61c.org/su22/resources/exams/

When posting questions, please reference the semester, exam, and question in this format so it's easier for students and staff to search for similar questions:

Semester-Exam-Question Number

For example: SP20-Final-Q1, or SU20-MT2-Q3

Summer 2020 midterm walkthrough

Fall 2020 midterm Bit Manipulations Walkthrough

Fall 2020 midterm 1 Slip Walkthrough

Alexander Ng

Su20-MT1-Q2b.

Why is 63 bytes of memory not freed and not 62 bytes? Wouldn’t 4 * 12 bytes for the nodes created + 13 bytes (for each character in *r) + 1 byte (for null terminator) be 62?

...
void free_list(node *n) {
    if (n == NULL) return;
    node *c = n->nxt;
    for (; c != n;){
        node *tmp = c; c = c->nxt;
        free(tmp);
    }
}

SU20-MT1-Q2. The solutions say that we are freeing the sentinel node in the stack only and not in
the heap. Is it because of the call to calloc is not accounted for?

Peyrin Kao  STAFF  3mth  #636bf
    I think this is an error in the solutions, sorry.

Te-Jung Chen  3mth  #636bc
    In summer 2020 midterm.
    Why is this -8192 to 8191? I though in the instructor's lecture he said branch should have an
extend zero is always odd and if we get 2 extra bit it should be [-4096 * 4 to 4094 *4] or [-2 ^ 14, 2
^ 14 -2]

Te-Jung Chen  3mth  #636ca
    The answer given is -2^13, 2^13-1

Te-Jung Chen  3mth  #636cb
    Is the answer wrong

Peyrin Kao  STAFF  3mth  #636be
    (edit: see below)

Te-Jung Chen  3mth  #636ca
    Sorry, let me explain this in a bit more detail: a standard branch instruction lets you
branch by 2^10 instructions in each direction. Since we get two more bits in the
immediate, we get 4 times as many instructions to branch to, so that's 2^12 instructions,
or 2^13 half-instructions in each direction.
Alexander Ng  3mth  #636cd
Replying to Peyrin Kao
Why does register fields only requiring 4 bits?
...

Te-Jung Chen  3mth  #636ce
Replying to Peyrin Kao
So you mean they are asking for instructions numbers not address. If it is instruction number should not have negative.
...

Peyrin Kao  STAFF  3mth  #636db
Replying to Te-Jung Chen
Register fields only require 4 bits because the question says there are only 16 registers in your new architecture.
The question is asking about the range of possible offsets, measured in half-word instructions.
...

Te-Jung Chen  3mth  #636dc
Replying to Te-Jung Chen
How do you know if they are asking instruction numbers or address. In Justin's lecture slide he literally specify the instructions range as address.
...

Peyrin Kao  STAFF  3mth  #636dd
Replying to Te-Jung Chen
The question says: “what is the range of half-word instructions that can be reached?”
Branch instructions use relative addressing to reach other instructions, so you wouldn't be able to provide a range of absolute addresses.
...

Te-Jung Chen  3mth  #636de
Replying to Peyrin Kao
...

Peyrin Kao  STAFF  3mth  #636df
Replying to Te-Jung Chen
Sorry, maybe I'm not fully understanding your question here. The slide is saying that with a 13-bit signed number, you can represent numbers in the range [-4096, 4094]. Then it converts that into $2^{10}$ instructions up/down.
...

Te-Jung Chen  3mth  #636ea
Replying to Te-Jung Chen
I can't tell if there asking the represent number in range or instructions count. They are both range for instructions just one is address one is number count.
Addresses are measured in bytes. The question asks you to measure the range in half-instructions, which are sets of 2 bytes.

I am confused over why the answer to b:

```c
fun->t *= -1; fun->t >>= 1;
printf("%d\n", fun->i[i]) is -1?
```

and

```c
fun->s[0] = '\0';
printf("%d\n", fun->t); is -256?
```

Does the original content of the Union get erased when we assigned to a variable of a different type with a larger size or is it retained?

Also, does the size here refer to the integer value size not the type size right?

Unlike structs, unions store all their components in the same space; we can imagine that while a struct is "Store an array of uints AND an array of ints AND an array of chars AND an int", a union is more "Store an array of uints OR ...". Each write tries to store in a specific part of the union, and ends up modifying the 32 bits of data that *fun is composed of. The reads then end up interpreting the same data in different ways.

How do you get ABCDD as the first 5 digits of the result?

auipc will add 0x100 to 0xABCDE000 to get 0xABCDE100. 0xABC will get sign extended to 0xFFFFFABC, which, when summed with 0xABCDE100 will give you 0xABCDDDBBC.
For this question, do we not translate the last bit from base three into decimal? Since the mantissa is 2222, with an implicit 1, I think we should translate $1.2222 \times 3^{15}$ into decimal as $1 \times 3^{15} + 2 \times 3^{14} + 2 \times 3^{13} + 2 \times 3^{12} + 2 \times 3^{11}$. However, the solution excludes the $2 \times 3^{11}$ term. Could you explain why we don't use the last bit in translation?

Peyrin Kao  
I think there was an error in the solutions here.

Anonymous Starling  
What is one's complement? Is that in scope?

Also, for the second question ($2+2 = \text{fish}$), what is the representation that would result in this being true?

Peyrin Kao  
One's complement is not in scope this summer.

2+2 = fish is not a representation that we've ever talked about. This question is just saying that you could hypothetically come up with a representation where "fish" represents 4 if you wanted to.

Anonymous Starling  
Would "addi a0, zero, 0" be the same as "li, a0, 0" in this context?

Peyrin Kao  
Yes, both put the value 0 in register a0.

Anonymous Starling  
Do you not put "typedef" before the word "union" in nested unions?

Peyrin Kao  
`typedef` is used if you want to give the union another name to refer to it.
For 5.b.A of this question, I am confused with why the answer is true. In my understanding, the range of biased number of n bits is \([-2^{n-1} + 1, 2^{n-1}]\), and the range of two's complement number of n bits is \([-2^{n-1}, 2^{n-1} - 1]\). As such, biased number should have one more positive than negative, and two's complement should have one more negative than positive. How can n bits of these two representations cover the same range of integers?

Peyrin Kao  STAFF  3mth  #636b

In bias notation, you can choose the bias to be whatever you want, unless it's something defined in the system (e.g. in floating point). The range you gave is for using a default bias of \(-2^{n-1}\), but you could also set the bias to be \(-2^{n-1} + 1\) which would give you the same range of numbers represented.