Review Lab Worksheet

1. Calling conventions

Consider this C code:

```
int test(int a, int b, int c) {
      int tmp = (a&b)+(a&b);
      int u = sum4(1,tmp,b,a);
      return u + a + tmp;
}
```

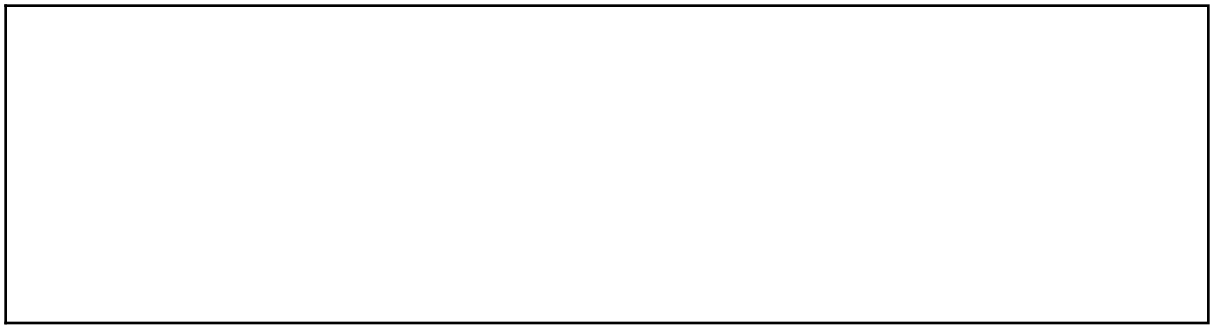
The body of the corresponding RISC-V assembly function is below.

```
      mv s1, a0
      mv s2, a1
      and t0, s1, s2
      add t1, t0, t0
      addi a0, x0, 1
      mv a1, t1
      mv a2, s2
      mv a3, s1
      sw t1, 0(sp)
      jal x1, sum4
      lw t1, 0(sp)
      add t0, s1, a0
      add a0, t0, t1
```

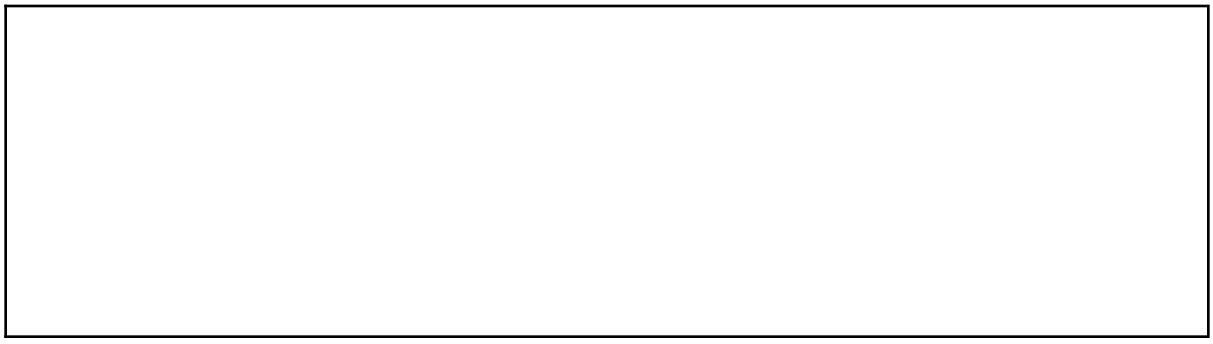
a. Identify the registers overwritten in this function body, and for each of them determine if their values need to be saved by the callee.

b. Write the function prologue and epilogue. Follow the RISC-V calling conventions that you learned in CS 3410.

```
.test:
```



```
...
[BODY]
...
```



2. Cache lookup

In a research computing system dedicated to measuring the performance of different cache systems, we have a direct-mapped cache with the following parameters:

- 8-bit addresses;
- block size of 4 bytes;
- four sets.

a. Compute the number of bits for the index, tag and offset.

b. Starting on an empty cache, diagram the following accesses: 0x04, 0x14, 0x18, 0x14. For each of them, compute their index, tag and offset, and determine if it is a hit or a miss, and the type of miss if applicable.

c. Thanks to a new grant, we add a 3-way associative cache with an LRU replacement policy. This cache is used as an L2 cache, with hit time 5 cycles. The original L1 direct-mapped cache has a hit time of 1 cycle, and accessing memory has a cost of 50 cycles. We still don't know what the miss rates are, so we may leave them as MR1 and MR2. Compute the overall average memory access time for a given access.

3. Virtual memory

a. Which of the following is false?

- ☐ Virtual addresses can be shorter than physical addresses.
- ☐ Virtual addresses can be longer than physical addresses.
- ☐ Two threads from the same process can both access the same page table.
- ☐ Two processes both calling `printf` can both access the same page table.
- ☐ Two processes both calling `printf` can both access the same code page in memory.

b. What are the advantages of using virtual memory instead of having programs directly access physical memory?

c. And the natural follow-up question: what are the disadvantages or costs of using virtual memory?