Last NAME: Shameem First Name: Azwad

Computer Science
C.Sc. 342/343

Quiz on Pipeline and Cache

CSc or CPE

Please submit to me as DM on slack by 6:15 PM. Thank you.

May 11, 2022

Please write your name on every page.
NO CORRECTIONS ARE ALLOWED !!!!! You may use back page for notes.
Please answer all questions. No computers are allowed.

Please hand write and sign statements affirming that you will not cheat:

"I will neither give nor receive unauthorized assistance on this exam. I will use only one computing device to perform this test"

Please hand write and sign **here:**

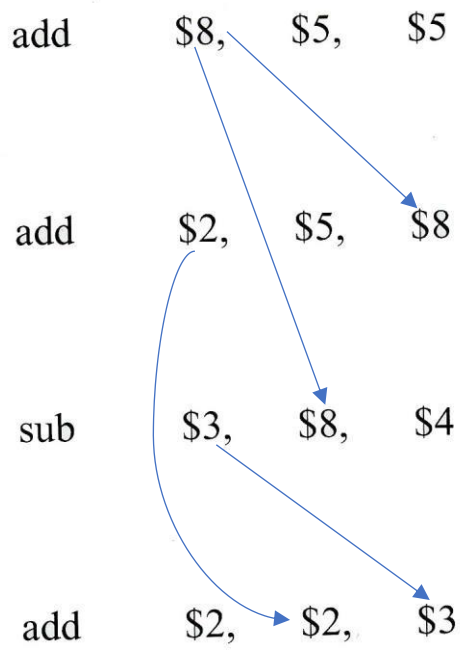
I will neither give nor recieve unathorized assistance on this exam. I will only use one computing device to perform this test

Azwad Shameem

Last NAME: Shameem First Name: Azwad

Question 1: (20 points) **Pipelined MIPS processor from the book.**

Show or list all of the dependencies in this program. For each dependency, indicate which instructions and register are involved.



You can draw the dependencies using ARROWS, or describe them in words:

Last NAME: Shameem First Name: Azwad

Question 2: (20 points) **Reordering Code to Avoid Pipeline Stalls**

Consider the following code segment in C:

```
A=B+E;
C=B+ F;
```

Here is the generated MIPS code for this segment, assuming all variables are in memory and are addressable using relative addressing mode i.e. using offsets from register $t0:

```
lw      $tl, 0($t0)
lw      $t2, 4($t0)
add     $t3, $tl,$t2
sw      $t3, 12($t0)
lw      $t4, 8($t0)
add     $t5, $tl,$t4
sw      $t5, 16($t0)
```

Find the hazards in the following code segment explain, and reorder the instructions to avoid any pipeline stalls.

The hazard is the add instructions that are dependent on the lw (load word) that comes before it, which causes a stall. However, it is possible to avoid this stalling issue by moving the third lw (load word) instruction before the add instructions.

```
lw      $tl, 0($t0)
lw      $t2, 4($t0)
lw      $t4, 8($t0)
add     $t3, $tl,$t2
sw      $t3, 12($t0)
add     $t5, $tl,$t4
SW      $t5, 16($t0)
```

By making the instructions in this order we can bypass the pipeline stall that was caused by the add instruction being dependent on the lw (load word) instruction.

Last NAME: Shameem First Name: Azwad

Question 3: (20 points) Cache computations

Average Memory Access Time is computed: AMAT = Hit time (Miss rate Miss penalty)

Assuming that *memory transfers take a total of 80 clock cycles.*

If the cache has a 95% hit rate and a one-cycle hit time, what is the average memory access time?

AMAT = Hit time + (Miss rate * Miss penalty)

AMAT = 1 + (0.05 * 80) = 5

Average Memory Access Time (AMAT) is 5 cycles.

Question 4: (20 points) What are the two characteristics of program memory accesses that caches exploit?

The first characteristic of memory access that caches exploit is **spatial locality**. Spatial locality is when the memory location is referenced and the locations with nearby addresses tend to be referenced soon.

The second characteristic of memory access that chances exploit is **temporal locality**. Temporal locality is when the memory location is referenced and then that location will tend to be referenced again soon.

Question 5: (10 points)

Please describe in one sentence what Cache miss is.

Cache miss is when the program attempts to access data from the cache, but the data is missing.

Question 6: (10 points)

Please describe in one sentence what is Cache hit.

Cache hit is when the program attempts to access data from the cache, and the data is found.