**MICROPROCESSOR AND COMPUTER ARCHITECTURE LABORATORY**

**UE19CS256**

**4th Semester, Academic Year 2020-21**

|  |  |  |
| --- | --- | --- |
| **Name:** Atul Anurag | **SRN:** PES2UG19CS075 | **Section:** B |

**Date: 21-04-2021**

Week#10

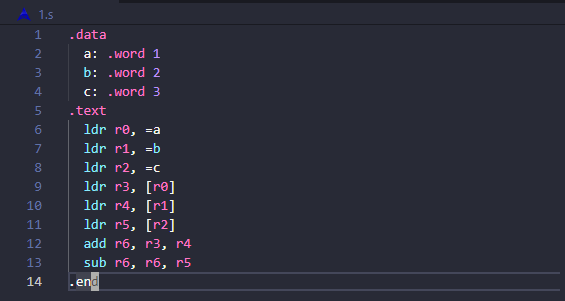
Program Number: \_\_\_\_1\_\_

**Given a C- Code convert it in its equivalent ARM Code.**

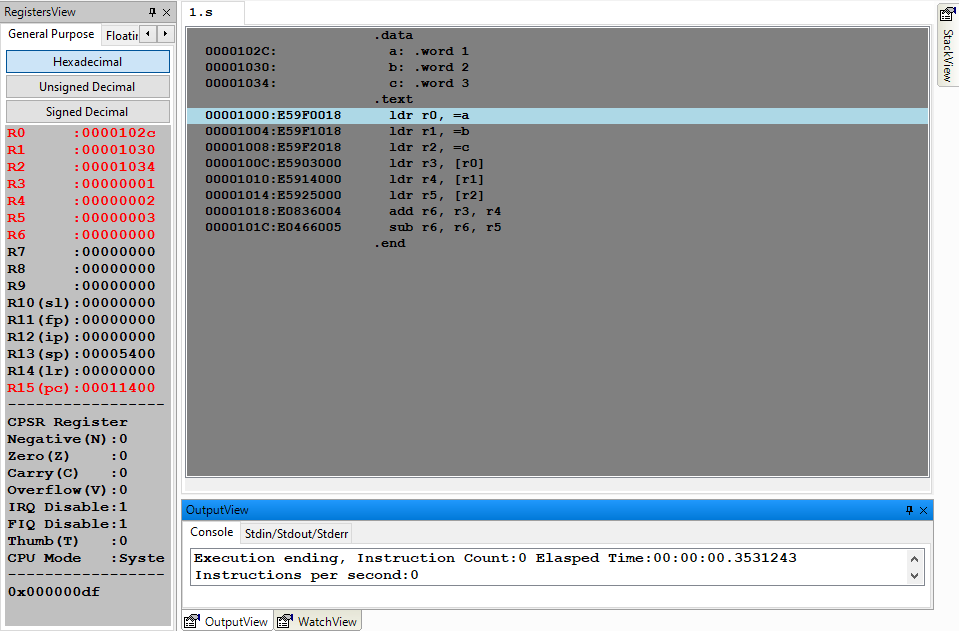
**These programs need to be executed on ARMSIM Simulator**

1. x = (a + b) - c

**ARM Assembly Language Code**

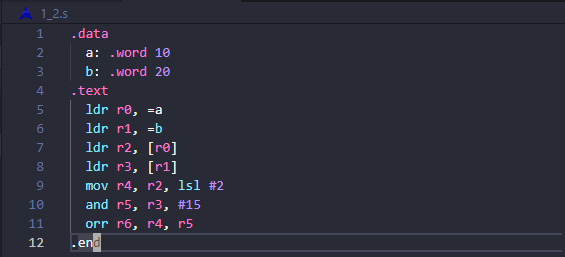
****

**Screenshot showing the value of x, a, b, c in the register window.**

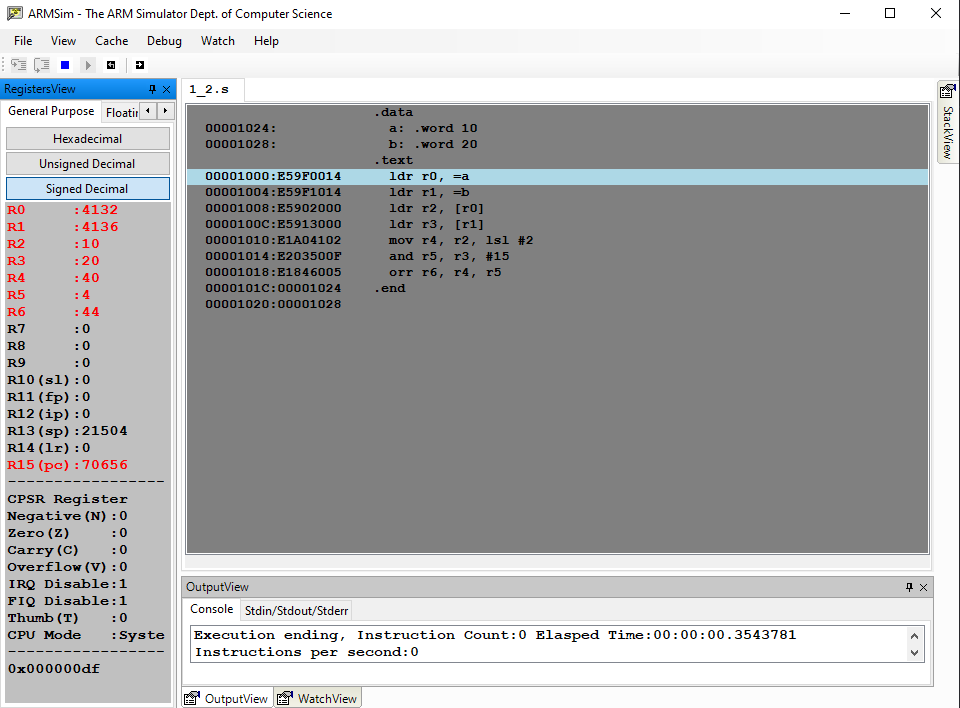
****

2) z = (a << 2) | (b & 15)

**ARM Assembly Language Code**

****

**Screenshot showing the value of a, b, z in the register window.**

****

Program Number: \_\_\_\_2\_\_

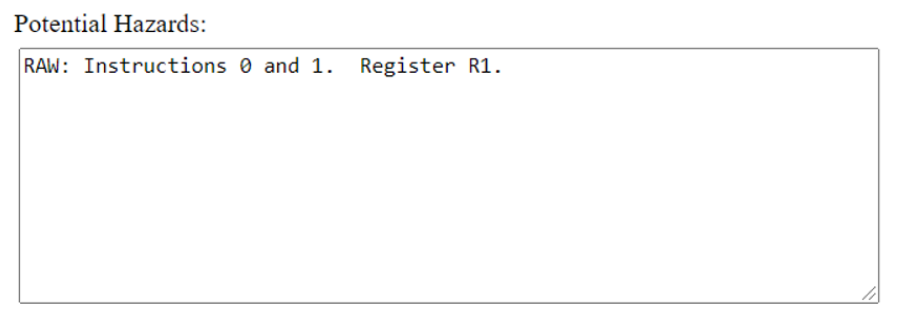
1) Consider the following instructions. Execute these instructions using simulator of 5 stage pipeline of MIPS architecture.

ADD R0, R1, R2

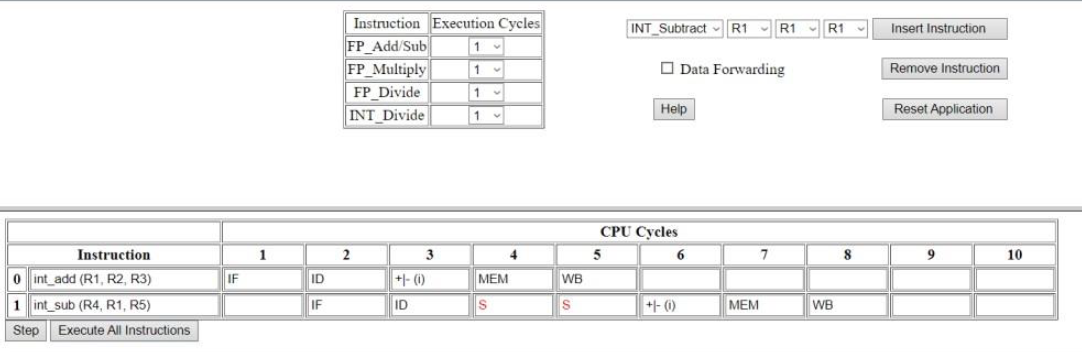
SUB R3, R0, R4.

Observe the following and note down the results.

1. Check whether there is data dependency for the second instruction?

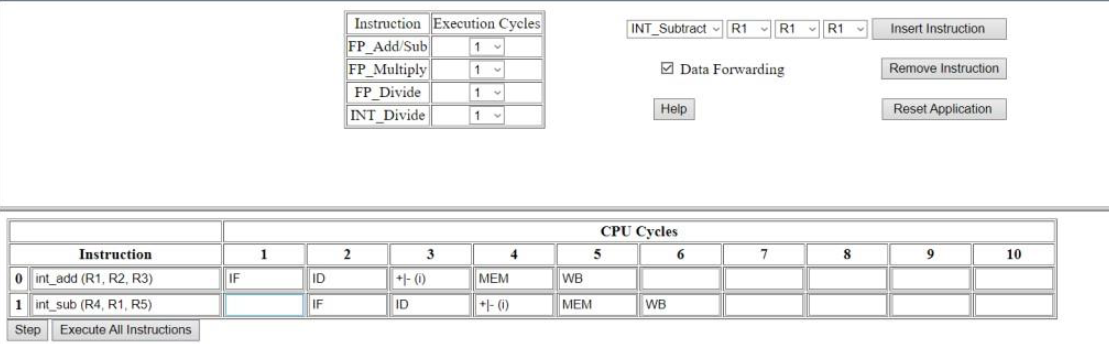


1. If yes, then, how many stall states have been introduced?



Without Data Forwarding, are 2 memory stalls.

1. If data forwarding is applied how many stall states have been reduced?



2 stalls have been reduced.

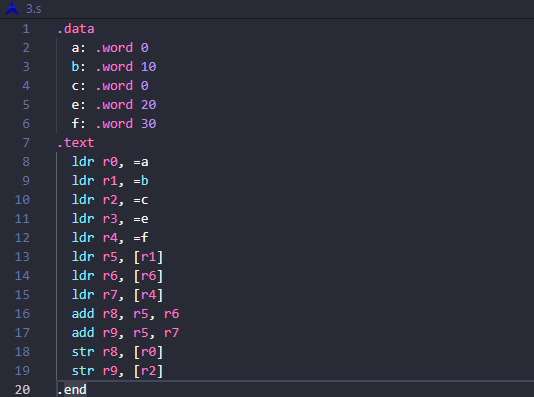
Program Number: \_\_\_\_3\_\_

Consider the following code segment in C.

A = B + E;

C = B + F;

1. Write the code using MIPS 5 STAGE pipeline architecture.

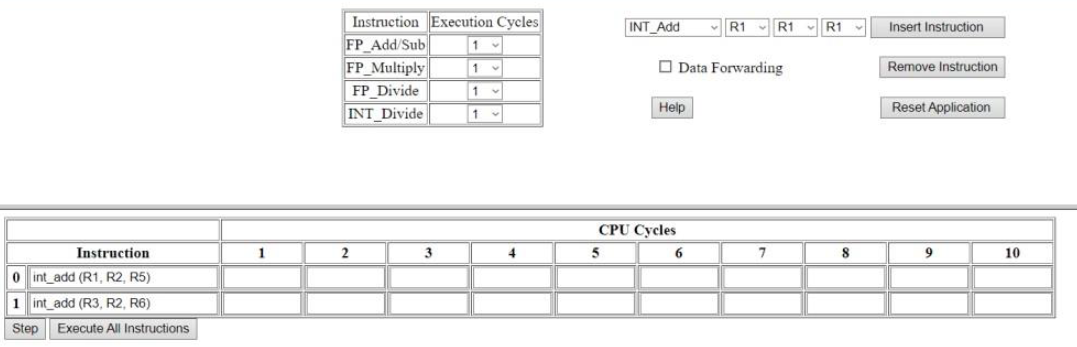
****

1. Find the hazards.



**No hazards found.**

c) Reorder the instructions to avoid pipeline stalls.



The instructions are already in order.

Program Number: \_\_\_4\_\_

Using MIPS 5 stage pipeline architecture, execute the following instructions and avoid stall states if any.

LW $10, 20($1)

SUB $11, $2, $3

ADD $12, $3, $4

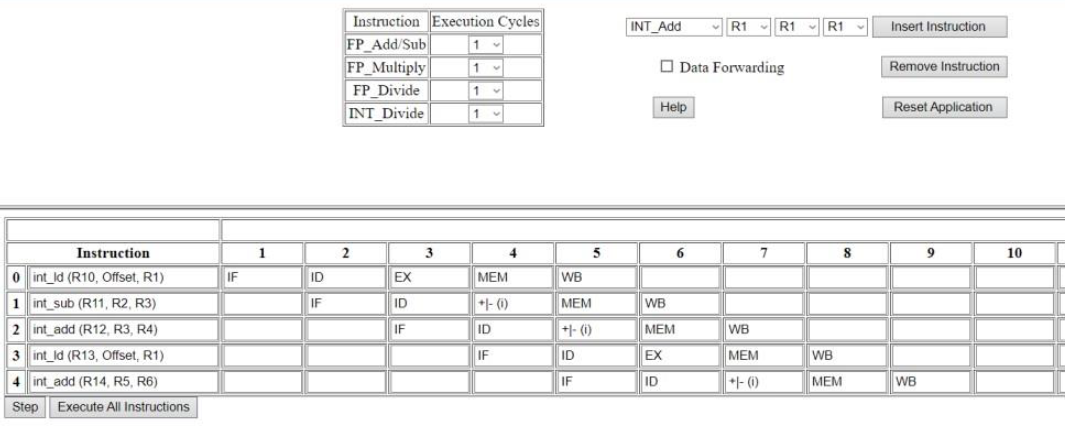
LW $13, 24($1)

ADD $14, $5, $6

**a) Related Screenshot with stalls**

**No stalls observed.**

**b) Related Screenshot without stalls**



Program Number: \_\_\_5\_\_

This exercise is to understand the relationship between delay slots, control hazards and branch execution in a 5 stage MIPS pipelined processor.

Label 1: LW $1, 40($6)

BEQ $2, $3, Label2 : branch taken

ADD $1, $6, $4

Label2: BEQ $1, $2, Label1 : branch not taken

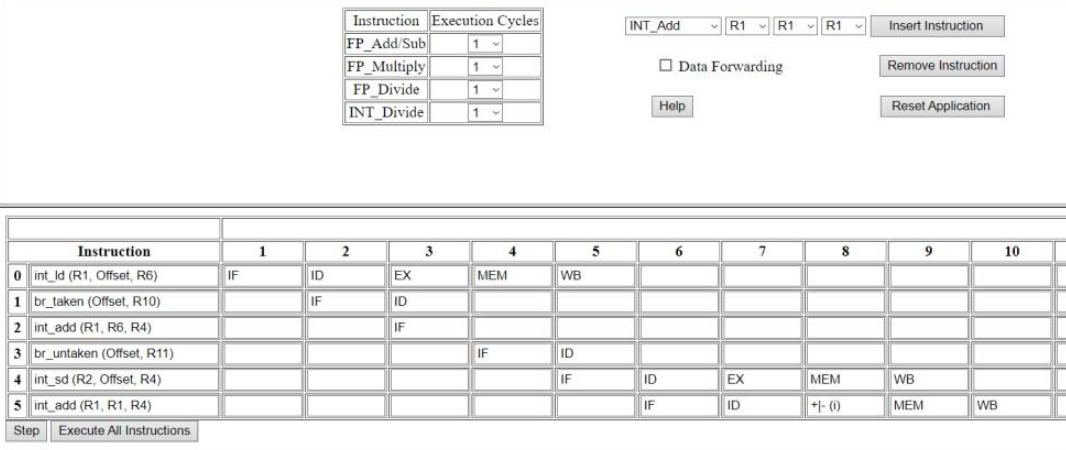
SW $2, 20($4)

ADD $1, $1, $4

Assume full data forwarding and predict- taken branch prediction.

Note the observations.

**Related Screenshot**



**Disclaimer:**

* The programs and output submitted is duly written, verified and executed by me.
* I have not copied from any of my peers nor from the external resource such as internet.
* If found plagiarized, I will abide with the disciplinary action of the University.

Signature: Atul Anurag

Name: Atul Anurag

SRN: PES2UG19CS075

Section: B

Date: 21-04-2021