Assignment 5
Report

Vishal Bindal
2018CS50425

Chirag Mohapatra
2018CS50403

- We implemented instructions bne, beq, blez, bgez, j, jal and jr in one clock cycle each
- In bne, beq, blez, and bgez the values in given registers are checked, and the program counter (pc) and addra are incremented accordingly
- In jal the value of current program counter is stored in $31 (return address) and then pc, addra are updated
- In jr, we have to jump to the instruction whose address is contained in the register, so pc and addra are updated accordingly.
- The stack pointer was kept in register $29 and initially set to 4095
- For non-leaf procedures, the return address has to be stored in the memory location stored in $29 and the stack pointer then updated

New test cases (to check the new instructions implemented:

Test case no.	Instruction (assembly)	Instruction(machine)	Final Output / details
1	sw $0 , 1024($1) sll $0 ,$0 , 2 sub $2 , $1 , $0 bgtz $2 , 2 lw $3 , 1017($0) blez $2 , 2 add $3 , $3 , $3 j 9 add $1 , $1 , $1	10101100001000000000010000000000, 00000000000000000000000010000000, 00000000001000000001000000100010, 00011100010000000000000000000010, 10001100000000110000001111111001, 00011000010000000000000000000010, 00000000011000110001100000100000, 00001000000000000000000000001001, 00000000001000010000100000100000,	2 (16 in $0 1 in $1 -14 in $2 2 in $3 2 in $4)

	jal 14 add $4 , $0 , $2 null null null add $2 , $2 , $2 sll $0 , 1 jr $31	00001100000000000000000000001110, 00000000000000100010000000100000, 00000000000000000000000000000000, 00000000000000000000000000000000, 00000000000000000000000000000000, 00000000010000100001000000100000, 00000000000000000000000001000000, 00000011111000000000000000001000;	
2	sll $4 , $0 , 3 jal 4 add $17 , $2 , $16 null sub $29 , $29 , $0 sw $31 , 1($29) sw $4 , 0($29) bgtz $4 , 3 add $29 , $29 , $0 jr $31 sub $4 , $4 , $1 jal 4 lw $4 , 0($29) lw $31 , 1($29) add $29 , $29 , $0	00000000000000000010000011000000, 00001100000000000000000000000100, 00000000010100001000100000100000, 00000000000000000000000000000000, 00000011101000001110100000100010, 10101111101111110000000000000001, 10101111101001000000000000000000, 00011100100000000000000000000011, 00000011101000001110100000100000, 00000011111000000000000000001000, 00000000100000010010000000100010, 00001100000000000000000000000100, 10001111101001000000000000000000, 10001111101111110000000000000001, 00000011101000001110100000100000,	Non leaf procedure Accumulator with n = 16 Assume 1 in $2 and 2 in $0 Argument in $4 $2 is $v0 Sp in $29 $17 contains final answer =136

	add $2 , $2 , $4 jr $31	00000000010001000001000000100000, 00000011111000000000000000001000;	
3	beq $0 , $1 , 5 sw $0 , 1024($1) sll $1 , $1 , 1 bne $0 , $1 , 2 beq $0 , $1 , 2 srl $1 , $1 , 1 lw $2 , 1023($1)	00010000000000010000000000000101, 10101100001000000000010000000000, 00000000000000010000100001000000, 00010100000000010000000000000010, 00010000000000010000000000000010, 00000000000000010000100001000010, 10001100001000100000001111111111;	2 (2 in $0, $1, $2)