Name ____

Register Name	Register Number	Usage
$zero	0	Constant 0
$at	1	Reserved for assembler
$v0, $v1	2, 3	Function return values
$a0 - $a3	4 – 7	Function argument values
$t0 - $t7	8 – 15	Temporary (caller saved)
$s0 - $s7	16 – 23	Temporary (callee saved)
$t8, $t9	24, 25	Temporary (caller saved)
$k0, $k1	26, 27	Reserved for OS Kernel
$gp	28	Pointer to Global Area
$sp	29	Stack Pointer
$fp	30	Frame Pointer
$ra	31	Return Address

The Execution Time Equation:

Execution time= Instructions * Cycles/Instruction * Time/cycle

Amdahl's Law:

$$\text{Time}_{\text{optimized}} = \text{Time}_{\text{unaffected}} + \text{Time}_{\text{affected}}/\text{optimizationFactor}$$

Baseline multicycle: Operations in each cycle; {B=3,R=4, SW=4, LW=5} cycles

Step	R-type	Memory	Branch

Instruction Fetch	IR = Mem[PC]
	PC = PC + 4		

Instruction Decode/ register fetch	A = Reg[IR[25-21]]
	B = Reg[IR[20-16]]
	ALUout = PC + (sign-extend(IR[15-0]) << 2)		

Execution, address computation, branch completion	ALUout = A op B	ALUout = A + sign-extend(IR[15-0])	if (A==B) then PC=ALUout

Memory access or R-type completion	Reg[IR[15-11]] = ALUout	memory-data = Mem[ALUout]
		or
		Mem[ALUout]= B	

Write-back		Reg[IR[20-16]] = memory-data	