



Research Project

Program on which the course is given	Computer Engineering and Software Systems
Course Code	CSE112
Course Title	Computer Organization and Architecture
Instructor	Dr. Tamer Mostafa
Deadline	3/6/2020
ILOs examined	Knowledge 1. Explain the MIPS instruction set architecture 2. Explain how computers execute programs 3. Illustrate how a data-path with or without pipelining is designed. 4. Infer the effect of the memory hierarchy on the performance Thinking skills 5. Write MIPS assembly programs 6. Translate C programs to assembly and assembly programs to machine code Subject-based practical skills 7. Design a processor's data path Skills for life and work (general skills) 8. Refer to relevant literature search for information and engage in life-long self-learning discipline. 9. Develop problem solving and creative thinking 10. Work and communicate effectively in team by effective collaboration and task management, working in a constrained stressful environment, and leading and motivating individuals.

The project consists of two parts:

Part-A) Designing and simulating a mini MIPS processor that should execute a specific code (40%)

Part-B) Designing and implementing an assembler and emulator that converts MIPS instructions into machine code and emulates its execution. (60%)

The project (two parts) is to be done in teams, each of a maximum of 5 students. The par of each student should be explicitly written in the report.

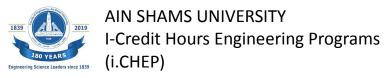


AIN SHAMS UNIVERSITY I-Credit Hours Engineering Programs (i.CHEP)



Part-A marking criteria

Part-A marking criteria																						
CSF112					Cours Name		nizati	uter ization and ecture			<u>Weight</u>		<u>40%</u>		Deadline		3/6/2020		<u>20</u>			
Student Name: ······															Student ID:							
		A (89)			Г		D (60	-66)			F (0-59)								
	100 96 92 88 88 88 80 76		75	72	69	67	99		64	62	09	59	40	20	c	>						
Completeness and creativity (30%)	Desconall to out	Design is complete and give all the required outputs with the best performance				Design is complete but does not give all the required outputs				Design is complete but gives wrong outputs				Design is particular complete				Design is w or missing r components				
Con																						
Simulation and results (40%)	• Simulation is complete and works efficiently					Simulation is complete but does not give all the required outputs				Simulation is complete but gives wrong outputs					ulatio tially o		olete	lete comp or mi			ulation is pletely wrong issing main ponents	
Simulat																						
Report (30%)	Except with language technology		 Good ability to express ideas with proper language and technical vocabulary. 				 Normal ability to express ideas with proper language and technical vocabulary. 				 Low ability to express ideas with proper language and technical vocabulary. 				S	 Difficult to exide as with principle and technical vocabulary. 			oper			
<u>~</u>																						
1 5	^{it} marl	cer To	otal		1 st marker S					gnatı	ıre					- -	ASU .			T		
2 ⁿ	^d Mar	ker T	otal			2 nd marker Sig						nature										1
General Comments:										UEL Grading Syste			stem Agreed Mark			k ASU Grading Scale				<u> </u>		
								9	% equivalent at UEL Rai					nge	% at AS			SU Grad		1		
								-		nd high		0/.	/			97% and higher			A+ A			
								_	82% to less than 95% 70% to less than 82%								93% to less than 97% 89% to less than 93%					
										66% to less than 70%								to less than 89%				
									_	63% to less than 66%						80% to less			ss than 84%			
									_	60% to less than 63% 56% to less than 60%								6 to less than 80%				
									_							73% to less tha 70% to less tha						
									_	53% to less than 56% 50% to less than 53%						67% to less t						
											45% to less than 50%						64% to less than 67%				D+	
										4		ess thar		%				to less t			D	
											Less than 40% Less than 60% F									F		





1. Part-A: Designing and simulating a mini MIPS processor that should execute a specific code

Requirements:

- Design and implement a program that shows the execution of the following specific code on a pipelined MIPS processor.

```
slti $at, $s5, 5
beq $at, $zero, Else
add $s6, $s5, $zero
j Exit
Else: add $s6, $zero, $zero
Exit:
```

- Use one of the following programming languages: any of the C family, Java, python.
- Provide screen shots of the emulated execution.
- The program should enable the user to trace the code execution line by line.
- The program should show the active stage of each instruction at each clock cycle.
- The input file could be a text file that contains the MIPS code.

Delivery procedure:

- Two files should be delivered in a folder named "CSE112_PA_G#", where G# is the group number:
 - PA_G#_report: The documentation of the project that follows the implementation procedure. After the title page, you should include a page declaring the role of each student in the project.
 - PA_G#_code:
 - o Source code of the program; fully documented.
- All files to be submitted in (.DOC, .DOCX) format ONLY.
- Project report to be Grammarly checked by student.
- Use at least 3 references and, at least, one of them must be an updated reference (after 2010). (for the two parts)
- Project report might undergo similarity checks for allocating plagiarism/cheating cases.

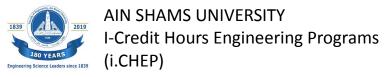


AIN SHAMS UNIVERSITY I-Credit Hours Engineering Programs (i.CHEP)



Part-B marking criteria

Part-B marking criteria																						
Course Code:	CS	(:SE112		Course Name:			Computer Organizatio Architecture					ight		60%	<u>.</u>	Deadline		3/6/2020		<u>0</u>		
Stude Nam														Student ID:		t						
	l	4 (89	-100)		В (76-88)					C (6	C (67-75)			D (60-66					F	F (0-59)		
	0	10 0				(0 10		~		7	ر .	,	4	~	0	0						
	100	96	92	89	88	84	80	26	75	72	69	67	99			62	09	59	40	20	0	
Completeness and creativity (30%)	Design is complete and give all the required outputs with the best performance				complete but does not give all the				con give	 Design is complete but gives wrong outputs 			Design comple			gn is partially plete		Design is we or missing n components			nain	
Con																						
mplementation and results (40%)		leme iplete ks ef		complete but does control cont				con give	plementation is nplete but es wrong puts			Implementatio partially compl				lete comp or mi			lementation is apletely wrong missing main apponents			
Implemer																						
Report (30%)	with lang tech	ress prop	ideas per e and I		Good ability to express ideas with proper language and technical vocabulary.				exp with lang tecl	Normal ability to express ideas with proper language and technical vocabulary.			 Low ability to express idea with proper language an technical vocabulary. 			deas er and	s idea lang d tech		as igua hni	icult to express as with proper guage and nnical abulary.		
Ř																						
1 s	^t mark	er To	otal		1 st mark				rker Sig	er Signature								greed ark				
2 nd Marker Total					2 nd marke					ignature												
General Comments:									_							rk ASU G		Grading Scale		e		
33									6 equiv	alent a	t UEI			% at AS				Grade				
											nd high		er			97% and high				Α+		
										2% to le			_			_	93% to less tha			Α		
										0% to le 6% to le					89% to less than 93% 84% to less than 89%			A- B+				
									_	66% to less than 70% 63% to less than 66%							80% to less than 84%			В		
										60% to less than 63%							76% to less th			B-		
										56% to less than 6						73% to less th				C+		
										53% to less than 56%				+			70% to less than 73%			С		
										50% to less than 53% 45% to less than 50%							% to less than 70% % to less than 67%			C- D+		
											40% to less than 45%						_	% to less than 64%			D	
											Less than 40%						Less than 60%				F	





Part-B: MIPS Simulator and Emulator

1- Requirements:

- Design and implement a program that converts MIPS assembly instructions into machine code, and emulates its execution.
- Use one of the following programming languages: any of the C family, Java, python.
- Provide screen shots of the emulated execution.
- The program should enable the user to trace the code execution line by line.
- A minimum of 19 instructions should be implemented. The instruction set should include:
 - o Arithmetic operations: ADD, ADDI, SUB,
 - o Logic operations: AND, ANDI, OR, ORI, NOR, SLL, SRL, SRA
 - o Compare instructions: STL, SLTI
 - Move instructions: LUILoad and store: LW, SW
 - Branch instruction: BEQ, BNE, J
- The input file could be a text file that contains the MIPS code.
- As a test to the program, you could show the implementation of the following code

```
slti $at, $s5, 5
beq $at, $zero, Else
add $s6, $s5, $zero
j Exit
Else: add $s6, $zero, $zero
Exit:
```

- You do not need to write a full program to be executed.
- You can assume the variables as memory locations that are pre-allocated.

2- Delivery procedure:

- Two files should be delivered in a folder named "CSE112_PB_G#", where G# is the group number:
 - PB_G#_report:
 - The documentation of the project that follows the implementation procedure.
 - Screen shots of the execution
 - After the title page, you should include a page declaring the role of each student in the project.
 - PB G# code:
 - Source code of the program; fully documented.



AIN SHAMS UNIVERSITY I-Credit Hours Engineering Programs (i.CHEP)



- All files to be submitted in (.DOC, .DOCX) format ONLY.
- Project report to be Grammarly checked by student.
- Use at least 3 references and, at least, one of them must be an updated reference (after 2010). (for the two parts)
- Project report might undergo similarity checks for allocating plagiarism/cheating cases.