Name	

Matrikel-nummer	

Exercise 1 (10 points)

Answer the following statements (Correct answer: +1 point; No answer: 0 points; Wrong answer: -1 point).

a) In 'Big-Endian', the left most byte is the most significant one.
Yes ☒ No ☐

b) The MIPS Register File contains sixty-four 32-bit registers.
Yes ☐ No ☒

c) For 'loads' immediately followed by 'stores', stalling can be avoided by forwarding.
Yes ☒ No ☐

d) 'Predict not taken' does not work well for 'top of the loop' branching structures.
Yes ☐ No ☒

e) The maximum bus speed is largely limited by the length of the bus and the number of devices on the bus.
Yes ☒ No ☐

Exercise 2 (5 Points)

a) (5 points) Suppose we have a processor with an ideal CPI of 1.5. The clock rate amounts to 1.6 GHz. A main memory access requires 40 ns. The proportion of load/store instructions amounts to 20% with a data cache failure rate of 4%. The failure rate for the instruction cache amounts to 3%.

i. (5 points) What is the overall CPI?

Your answer:

CPI = 1.5 (cycle) + .2 (prop of load/store) x 0.04 (failure rate) x 40ns (memory delay) x 1.6 GHz (clock cycles) + 0.03 (failure rate to load any instruction) x 40ns (memory delay) x 1.6 GHz (clock cycles) = 1.5+0.384+1.44=3.32

Exercise 3 (10 Points)

Suppose a MIPS processor with the simple 5-stages (IF, ID, EX, MEM and WB) pipeline is given. The instruction memory and data memory are separate and the register file can support one simultaneous read/write operation in one clock cycle. The processor supports forwarding only from ALU to ALU. In the case of the branch instruction, the next instruction to feed into the pipeline becomes known when the execute stage of the branch instruction has completed. Branch prediction is not supported. In the absence of hazards, a new instruction can be fed to the pipeline every cycle.

Given the following MIPS Code.

```
        1 add  $t1, $t2, $t3
        2 lw   $s2, 200($t1)
        3 add  $s3, $t1, $s2
        4 beq  $s3, $s3, L1
        5 sw   $s2, 200($t1)
L1:     6 addi $s1, $s1, 4
```

i. (10 points) How many cycles does this code take to complete? Show in a diagram the schedule in which the stages of the instructions in the above MIPS code are executed. Also indicate explicitly where forwarding or a simultaneous R/W operation can be used to resolve hazards.

Your answer:

	1	2	3	4	5	6	7	8	9	10	11	12	13
add	IF	ID	EX	M	RW								
lw		IF	ID	EX	M	RW							
add			*	*	IF	ID	EX	M	RW				
beq				*	*	IF	ID	EX	M	RW			
L1:addi									IF	ID	EX	M	RW

Exercise 4 (10 points)

a) (10 points) Given the following code fragment

```
do{
    g=g+A[i];
    i=i+j;
  } while (i != h)
```

where `A[]` is an array of integers, translate the code fragment to MIPS Assembly code. Assume that the variable `g` is in `$s1, h` is in `$s2, i` is in `$s3, j` is in `$s4 and 'base address of A'` is in `$s5.`

Your answer:

```
LOOP: sll  $t1, $S3, 2          # $t1=4*i
      add  $t1, $t1, $S5        # $t1=addrA
      lw   $t1, 0($t1)          # $t1=A[i]
      add  $S1, $S1, $t1        # $g=g+A[i]
      add  $S3, $S3, $S4        # i=i+j
      bne  $S3, $S2, LOOP       # go to LOOP if i!=h
```

Exercise 5 (4 points)

Answer the following questions:

a) (2 points) What are the definitions for latency and bandwidth?

Your answer:

Latency: time to access on word.
Bandwidth: how much data from the memory can be supplied to the processor per unit time.

b) (2 point) In MIPS unconditional branch instructions, how is the destination address specified?
(BIT FORMAT)

Your answer:

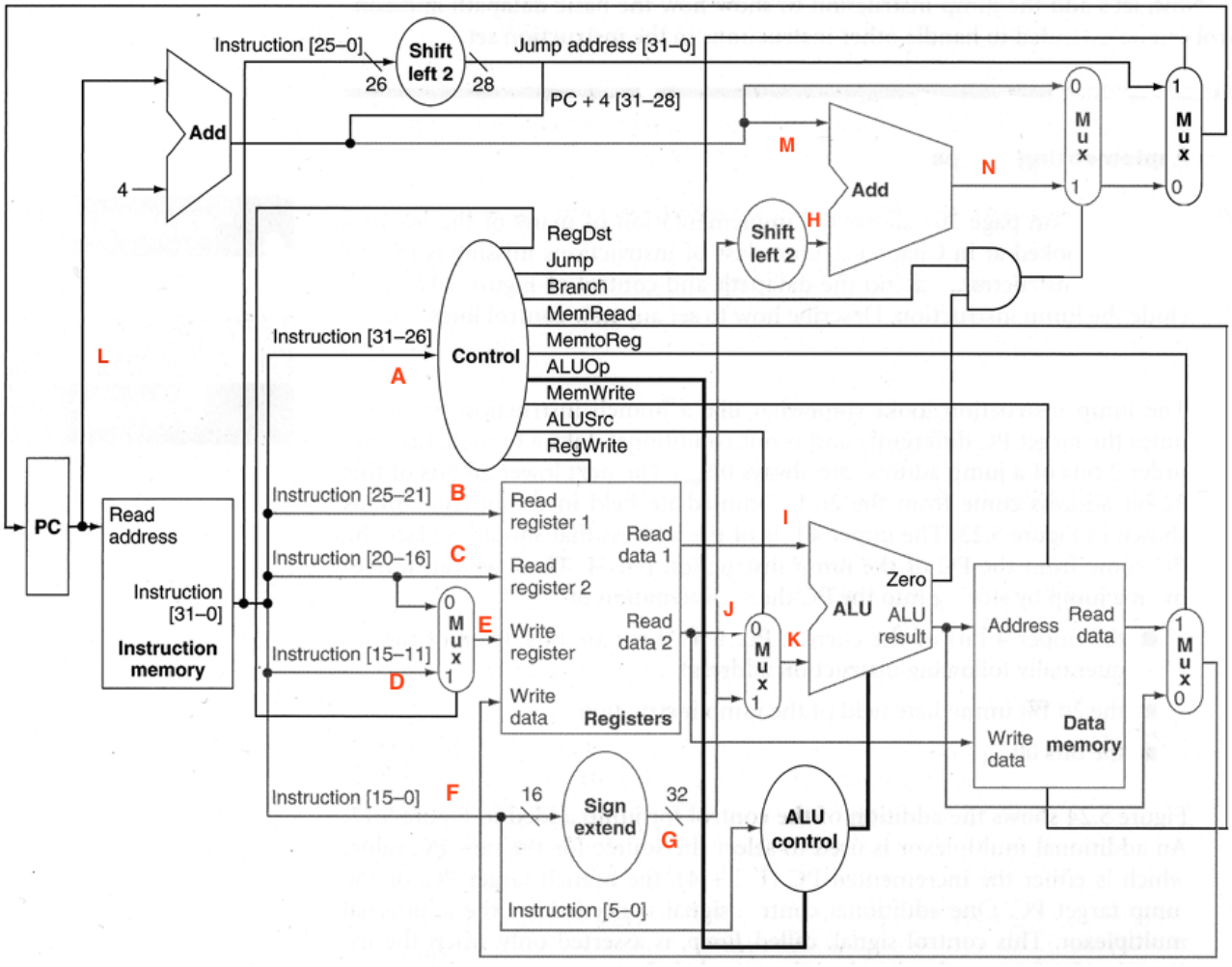
Syntax:	j target
Encoding:	0000 10ii iiii iiii iiii iiii iiii iiii

j label; |op|26-bit| 4 from PC + 2 zeros shift Slided V03 page 20

Exercise 6 (12 points)

You are given the following single cycle implementation of the MIPS ISA. Assume that a program (at instruction memory address `0x0000'2024`) executes the instruction `beq r20,r23,-0x7`. The state of the register file is given in the following table:

Register file	
Register	**Content**
`PC`	`0x0000'2024`
`$r20`	`0x0000'001C`
`$r21`	`0x0000'1008`
`$r22`	`0x1004'006A`
`$r23`	`0x0000'001C`



Note: For this exercise the `0x` prefix is used to indicate hexadecimal values.

Based on the given implementation and register states, answer the following questions:

a) (5 points) How is the instruction `beq r20,r23,-0x7` encoded? State the binary representation of this instruction and document your solution process. The opcode of `beq` is `0x04`.
(Hint: `beq` is an I-type instruction, the syntax of an `beq` instruction is `beq $rs,$rt,offset`)

Your answer:

```
opcode = 4 = 0x04 = 0b0000'0100
rs = 20 = 0x14 = 0b0001'0100
rt = 23 = 0x17 = 0b0001'0111
immediate = offset = -7 = 0xFFF9 = 0b1111'1111'1111'1001
(7 = 0b0000'0000'0000'0111, 7 = 0b1111'1111'1111'1001)

I-Type:    opcode(6)|rs(5)|rt(5)|immediate(16)
Result:    000100|10100|10111|1111'1111'1111'1001
           0x04|0x14|0x17|0xFFF9
           4|20|23|-7/65529(unsigned)
```

b) (7 points) For each letter in the diagram (A…Q), state the value of the signal on the corresponding wire / bus. You may use binary, hexadecimal or decimal values. All (if any) undefined signals are to be marked with `X`.
Additionally, what are the values of the control signals `RegWrite, Branch` and `Jump`?

Your answer:

A	**B**	**C**	**D**	**E**
0x04 (op)	0x14 (rs)	0x17 (rt)	0b1'1111	X
F	**G**	**H**	**I**	**J**
0xFFF9 (imm)	0xFFFF'FFF9 (s.e. imm)	0xFFFF'FFE4 (G << 2)	0x1C r[20]	0x1C r[23]
K	**L**	**M**	**N**	
->J	0x2024 (pc)	0x2028 (pc+4)	0x200C (pc+4+(-7)*4)	

Exercise 7 (6 points)

a) (2 points) Explain the meaning of the following variable declarations:
   i. `int *ptr[30];`
   ii. `int *a, b;`

Solution:

i) ptr is an Array of 30 integer pointer.
ii) a is a pointer to integer, b is an integer

b) (4 points) What is the output of the following program?

```
##include<stdio.h>

#define CUBE(x) (x*x*x)

int cube_fun(x){

   return x*x*x;
}

int main(void) {
    int a = 3;
    int b = CUBE(a++);
    printf("%d, %d\n", a, b);
     int c = cube_fun(a++);
   printf("%d, %d\n", a, c);

    return 0;
}
```

Your answer:

6, 60
7, 216