Department of Electrical and Computer Engineering
The University of Texas at Austin

EE 360N, Spring, 2001
Yale Patt, Instructor
Onur Mutlu, Kameswar Subramaniam, TAs
Final Exam, May 14, 2001

Name:_____

Problem 1 (20 points):_____

Problem 2 (20 points):_____

Problem 3 (10 points):_____

Problem 4 (15 points):_____

Problem 5 (10 points):_____

Problem 6 (10 points):_____

Problem 7 (15 points):_____

Total (100 points):_____

Note: Please be sure that your answers to all questions (and all supporting work that is required) are contained in the space provided.

Note: Please be sure your name is recorded on each sheet of the exam.

GOOD LUCK!

Name:_____

Problem 1 (20 points):

Part a (3 points): p% of a program is vectorizable. We wish to run it on a multiprocessor. Assume we have an unlimited number of processing elements. If the maximum speedup achievable on this program is 100, what is p?

Part b (5 points): The fundamental distinction between interrupts and exceptions is that interrupts are caused by _____ and exceptions are caused by _____.

Interrupts are handled **mostly** when convenient. Why?

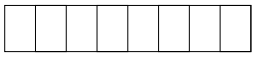
Why are interrupts not **always** handled when convenient?

Part c (6 points): In most of the Snoopy cache protocols, each cache block is in one of four states, depending on what information the protocol felt was important to capture. What information is captured by the Illinois protocol that is not captured by the Goodman protocol?

What benefit does this information yield?

Part d (6 points): Assume IEEE Floating point, adapted to accommodate a 8-bit representation (1 sign bit, 4 exponent bits, using an excess-7 code, 3 fraction bits). Can the number 23 be represented exactly in this 8-bit format? Yes / No (Circle one)

If yes, show the representation:



If no, show the two possible representations, depending on the rounding mode:

Name:_____

Problem 2 (20 points):

We have a byte-addressible toy computer that has a physical address space of 512 bytes. The computer uses a simple, one-level virtual memory system. The page table is always in physical memory. The page size is specified as 8 bytes and the virtual address space is 2 KB.

Part A.

i. (1 point) How many bits of each virtual address is the virtual page number?

ii. (1 point) How many bits of each physical address is the physical frame number?

We would like to add a 128-byte write-through cache to enhance the performance of this computer. However, we would like the cache access and address translation to be performed simultaneously. In other words, we would like to index our cache using a virtual address, but do the tag comparison using the physical addresses. The cache we would like to add is direct-mapped, and has a block size of 2 bytes. The replacement policy is LRU. Answer the following questions:

iii. (1 point) How many bits of a virtual address are used to determine which byte in a block is accessed?

iv. (1 point) How many bits of a virtual address are used to index into the cache?

v. (6 points) What is the size of the tag store in bits?

Name:_____

Problem 2 continued:

Part B.

Suppose we have two processes sharing our toy computer. These processes share some portion of the physical memory. Some of the virtual page-physical frame mappings of each process are given below:

PROCESS 0	
Virtual Page	Physical Frame
Page 0	Frame 0
Page 3	Frame 7
Page 7	Frame 1
Page 15	Frame 3

PROCESS 1	
Virtual Page	Physical Frame
Page 0	Frame 4
Page 1	Frame 5
Page 7	Frame 3
Page 11	Frame 2

vi. (2 points) Give a complete physical address whose data can exist in two different locations in the cache.

vii. (2 points) Give the indexes of those two different locations in the cache.

viii. (6 points) We do not want the same physical address stored in two different locations in the 128-byte cache. We can prevent this by increasing the associativity of our virtually-indexed physically-tagged cache. What is the minimum associativity required?

Name:_____

Problem 3 (10 points):

Complete the truth table for modified Booth's algorithm in which up to 3 bits are consumed in one cycle. Some of the entries are already completed for you.

Bits	Carry	pre_shift	ALU_op	post_shift	carry
000	0	3	PASS	0	0
001	0				
010	0				
011	0				
100	0				
101	0				
110	0				
111	0	0	SUB	3	1
000	1				
001	1				
010	1				
011	1				
100	1				
101	1				
110	1				
111	1				

Using the truth table you drew above, we ask you to multiply 011100111 by 011100111. How many cycles are required to finish this operation: [ ]

Show the contents of the temporary result register after each step:

	0	1	1	1	0	0	1	1	1							
	0	1	1	1	0	0	1	1	1							
After step 1:																
After step 2:																
After step 3:																
After step 4:																
After step 5:																

Name:_____

Problem 4 (15 points):

A number is represented with 32 bits, using IEEE Floating Point format. (1 sign bit, 8 exponent bits, 23 fraction bits). Your job: design a piece of logic that can quickly multiply this number by 2. There are five distinct cases that you have to consider. In the box provided, identify the five cases.

Case 1: _____

Case 2: _____

Case 3: _____

Case 4: _____

Case 5: _____

For each of these cases, in 20 words or fewer, describe what the logic must do.

Case 1:

Case 2:

Case 3:

Case 4:

Case 5:

Name:_____

Problem 5 (10 points):

```
for ( i = 0; i < 64; i++ )
    A[i] = A[i] + k
```

We would like to execute the above loop in two different processors (k is a scalar).

a. Write the Cray-like assembly code for this loop. How many cycles would it take on a vector processor with chaining? Assume that the vector processor has 1 read port and 1 write port to memory. The memory is 64-way interleaved. Vector registers are all 64 elements long. Loads and stores take 11 cycles, the add operation takes 4 cycles, and all functional units are fully-pipelined.

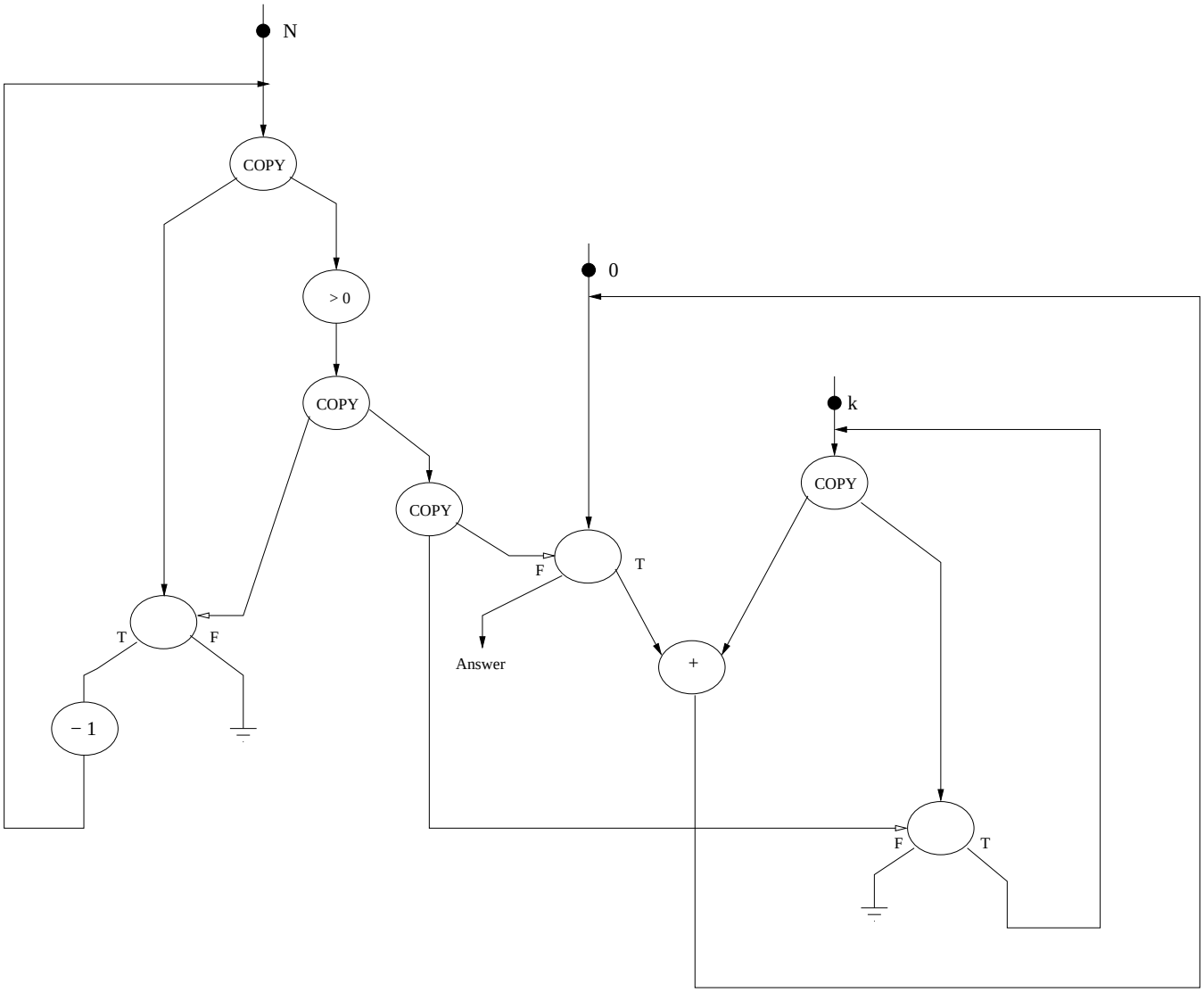
b. We would like to execute the same loop on an array processor using the same memory. You are free to use as many functional units as needed. What is the minimum number of cycles it would take to execute the same loop on an array processor? (Assume the same latencies for load, store, and add operations)

c. What is the speedup obtained from using the array processor over the vector processor?

Name:_____

Problem 6 (10 points):

The following data flow graph requires three non-negative integer input values 0 (zero), N, k as shown. It produces a value "Answer", also shown.



What function does this data flow graph perform (in less than ten words, please)?

Name:_____

Problem 7 (15 points):

You are asked to implement a new instruction, PUSH, in the LC-2, which pushes one value onto the stack. The specification of this instruction is as follows:

Assembler Formats

PUSH SR

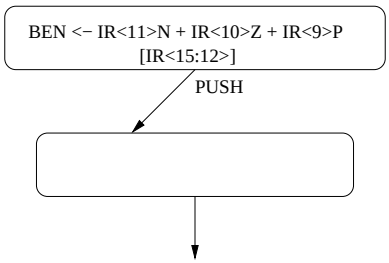
Encodings

Operation

R6 = R6 + 1;
Memory[R6] = SR;

a. To support the implementation of this instruction we have added to the datapath a register that is hardwired to constant 1. You should use the ALU to increment R6 in the implementation of the PUSH instruction. You are allowed to add any other control signals and structures you may need.

Draw the additional states you need to implement PUSH. You are allowed to add as many states as you need to the state diagram. ('A' students will not need to use the constant 1 or any other new registers, or add any new signals to the microcode). Using the notation of the LC-2 state diagram, shown on page 11, describe what happens in each state. The bubble for the first additional state is shown for you.



Name:_____

Problem 7 continued:

b. Describe the changes in the datapath you need to make to implement PUSH. List the new control signals that you need to add to the control store, if any. In addition, show these changes on the datapath (which is included on page 12).