Andrew ID: ____________________
Full Name: ____________________

Midterm Exam
Time Limit: 150 Minutes

Instructions

- This exam contains 9 pages (including this cover page) and 6 questions.
- This exam is an individual effort.
- You are not permitted to help others, in any way, with this exam.
- You are not permitted to release or to discuss this exam with anyone, except the course staff, until given permission to do so by the instructors (which will not occur until all students have completed the exam. There may be exceptional cases that take it late).
- You are permitted to use only the official course textbook, the official course slides, and your own personal notes.
- A simple calculator is permitted, but won't prove to be helpful (we don't think).
- You have 120 minutes, from first exposure through submission to take this exam. Do not attempt to "peek", "check", or "test" the exam. This will start your clock.
- We only expect the exam to take 70-90 minutes.

Question	Topic	Points
1	Integers	15
2	Floats	15
3	Assembly	15
4	Structs and Alignment	15
5	Caching	20
6	VM	20
Total:		100

1. (15 points, 3 points per blank) **Integers**

This question is based upon the following declaration on a **machine using 6-bit two's complement arithmetic for signed integers.**

Fill in the empty boxes in the table below.

- **Show all digits for the "Binary" column, including any leading 0s.**
- You need not fill in entries marked with "–".
- TMax denotes the largest positive two's complement number
- TMin denotes the smallest negative two's complement number.
- Hint: Be careful with the promotion rules that C uses for signed and unsigned ints, i.e. how the C Language handles implicit casts between the types.

Expression	**Decimal Representation**	**Binary Representation**
-	-5	(A)
-	(B)	001010
Tmin	(C)	-
-	-1	(D)
--	-16	(E)

2. (15 points) **Floats**

This question is based upon an IEEE-like floating point format with the following specification:

- 6-bit width
- There is s = 1 sign bit
- There are k = 3 exponent bits
- There are n = 2 fraction bits
- Wherever rounding is necessary, round-to-even should be used. In addition,you should give the rounded value of the encoded floating point number.

(12 points, 3 points each) Fill in the following:

Value	**Binary Representation**	**Rounded Value**
5	(A)	(B)
$-\frac{1}{16}$	(C)	--
$\frac{1}{32}$	(D)	--

(3 points, 1.5 point each) How many numbers could be represented using this representation in thefollowing ranges?

Range	Count of numbers
Normalized	(E)
Denormalized	(F)

3. (15 points) **Assembly**

Please consider the following assembly code:

```
loopquestion:
        cmpl    %esi, %edi
        movl    %esi, %eax
        cmovle  %edi, %eax
        cmpl    %edx, %eax
        jge     .L1
        cmpl    %edx, %edi
        jge     .L3
.L4:
        addl    $1, %edi
        cmpl    %edi, %edx
        jne     .L4
.L3:
        cmpl    %esi, %edx
        jle     .L1
.L5:
        addl    $1, %esi
        cmpl    %esi, %edx
        jne     .L5
.L1:
        ret
```

(A) (3 points) How many loops are within this question?

(B) (3 points) How many if statements are within this question (that can't be considered part of the pre-test for an if or while loop)?

(C) (3 points) Do two or more loops share the same loop control variable (a variable which is updated by the body of the loop and used as part of the test for the loop)?

(D) (3 points) Do two or more loops share the same end point? In other words, do they stop when the loop control variable reaches the same value or condition?

(E) (3 points) Where do the data values for the loop(s)' starting and stopping conditions come from?

a. Function arguments
b. Constants
c. Runtime inputs
d. More than one of the above
e. None of the above

4. (15 points) **Structs and Alignment**

Consider the following struct:

```
struct {
  short s; // 2-byte type
  int i;   // 4-byte type
  char c;  // 1-byte type
} exam;
```

Assume a system which requires “natural alignment”, i.e. each type needs to be aligned to a multiple of its size (width).

(A) (3 points) How many bytes of padding would the compiler place immediately before s?

(B) (3 points) How many bytes of padding would the compiler place immediately before i?

(C) (3 points) How many bytes of padding would the compiler place immediately before c?

(D) (3 points) How many bytes of padding would the compiler place immediately after c?

(E) (3 points) Given the declaration, “`struct exam array[4]`”, what would be returned by “`sizeof(array)`”?

5. (20 points) **Caching and the Memory Hierarchy**

Part A: Caching (10 points)

Given a model described as follows:

- Number of sets: 4
- Total size: 48 bytes (not counting meta data)
- Block size: 4 bytes/block
- Replacement policy: Set-wise LRU
- 8-bit addresses

5(A) (2 point) How many bits for the block offset?

5(B) (2 point) How many blocks per set?

5(C) (2 point) How many bits for the tag?

(8 points, 1 point each): For each of the following addresses, please indicate if it hits, or misses, and if it misses, if it suffers from a capacity miss, a conflict miss, or a cold miss:

Question Number	**Address**	**Circle one (per row):**
5(D)	0xA0	Hit Miss Unknowable
5(E)	0xAE	Hit Miss Unknowable
5(F)	0x52	Hit Miss Unknowable
5(G)	0XFC	Hit Miss Unknowable
5(H)	0XC2	Hit Miss Unknowable
5(I)	0XA2	Hit Miss Unknowable
5(J)	0XA0	Hit Miss Unknowable
5(K)	0X50	Hit Miss Unknowable

Part B: Locality (6 points)

Consider the following code:

```
int array[SIZE1][SIZE2];
int sum=0;
for (int outer=0; outer<SIZE1; outer+=STEP)
   for (int inner=0; inner<(SIZE2-1); inner++)
     sum += array[inner][outer] + 2*array[inner][outer+1];
```

5(L) (3 points): Imagine arrays extremely large in all dimensions, an int size of 4 bytes, and a cache block size of 16 bytes. To the nearest whole percent or simple fraction, what would you expect the combined miss rate for accesses to "array" and "array2" within the inner loop to be?

Part C: Memory Hierarchy and Effective Access Time

Imagine a system with a main memory layered beneath a cache:

- The cache has a 5nS access time.
- The main memory has an access time of 10nS.
- The cache miss rate is 1%.
- In the event of a miss, memory access time and cache access time do not overlap: They occur 100% sequentially, one after the other.

5(M) (3 points) Which of the following is closest to the effective, overall access time?

EFFECTIVE_ACCESS_TIME=

6. (20 points) **Virtual Memory**

Consider a virtual address system with the following parameters:

- The memory is byte addressable.
- Virtual addresses are 16 bits wide.
- Physical addresses are 16 bits wide.
- The page size is 1024 bytes.
- The TLB is fully associative with 16 total entries.
- *Hint*: Recall that a fully associative cache has just one set of entries.

In the following tables, all numbers are given in hexadecimal. The contents of the TLB and the page table for the first 16 virtual pages are as follows. If a VPN is not listed in the page table, assume it generates a page fault.

TLB

Tag	PPN	Valid
03	1B	1
06	06	0
28	23	1
01	18	0
31	01	1
12	00	0
07	3D	1
0B	11	1
2A	2C	0
11	1C	0
1F	03	1
08	14	1
09	2A	1
3F	30	0
10	0D	0
32	11	0

Page Table

VPN	PPN	Valid
00	27	1
01	0F	1
02	19	1
03	1B	1
04	06	0
05	03	1
06	06	0
07	3D	0
08	14	1
09	2A	1
0A	21	1
0B	11	1
0C	1C	1
0D	2D	0
0E	0E	0
0F	04	1

(a) (2 points) What is the maximal physical memory size in bytes (decimal)? _______

In the three rows below, mark the bits that constitute the indicated part of the virtual address with a Y and the other bits with an N. Enter your solution as a string, e.g. NNNNNNNNNNNNYYYY

(b) Virtual Page Number (2 points)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VPN																

(c) Virtual Page Offset (2 points)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
VPO																

(d) TLB Tag (2 points)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TLBT																

(e) Rewrite the virtual address 0x05DD in binary using only 1s and 0s, e.g. 0000000011111111 (2 points)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Please complete the following table for virtual address 0x05DD (2 points each)

Parameter	Value
6(F) VPN (hex)	
6(G) TLB Tag (hex)	
6(H) TLB Hit? (Y/N)	
6(I) Page Fault? (Y/N)	
6(J) PPN (hex)	