THE AUSTRALIAN NATIONAL UNIVERSITY

Mid Semester Examination, April 2009

COMP2300 / COMP6300
(Introduction to Computer Systems)

Writing Period: 1 hour duration

Study Period: 0 minutes duration

Permitted Materials: One A4 page with notes on both sides.
NO calculator permitted.

Questions are NOT equally weighted.
This exam will contribute 20% (redeemable) to your final assessment.

The questions are followed by labelled, framed blank panels into which your answers are to be written. Additional answer panels are provided (at the end of the paper) should you wish to use more space for an answer than is provided in the associated labelled panels. If you use an additional panel, be sure to indicate clearly the question and part to which it refers to.

The marking scheme will put a high value on clarity so, as a general guide, it is better to give fewer answers in a clear manner than to outline a greater number in a sketchy, half-answered fashion. The Appendix contains a table with powers of 2 values in decimal.

Please write clearly – if we cannot read your writing you may lose marks!

Student Number:

Official use only:

Q1 (13)	Q2 (17)	Total (30)

QUESTION 1 [13 marks]

(a) Assume memory addresses `0x02000411` to `0x02000415` contain the following 8-bit binary values:

Address	`0x02000411`	`0x02000412`	`0x02000413`	`0x02000414`	`0x02000415`
Binary value	`0010 1011`	`1001 0011`	`1100 1010`	`0101 0111`	`0001 0110`

Assume `sizeof(x)` and `sizeof(y)` are both 2, `&x = 0x02000412` and `&y = 0x02000414`, and the data storage is *little endian*.

(i) What would be printed by the following C statement?

```
printf("Values for x+y: %x %o\n", x+y, x+y);
```

Clearly show how you derive your answers.

QUESTION 1(a)[i] **[2 marks]**

(ii) What would be printed by the following C statement?

```
printf("Value for (x+y)/256: %d\n", (x+y)/256);
```

Clearly show how you derive your answer.

QUESTION 1(a)[ii] **[1 mark]**

Question 1 (continued)

(b) The IEEE single-precision floating-point standard is: 1 bit sign, 8 bits exponent with a bias of 127, and the remaining 23 bits are the mantissa (with an implicit leading bit). What floating point number is represented by the following 32-bit number (given in hexadecimal representation):

`0x41B80000`

For full marks, your answer should be expressed as a single decimal number accurate to four decimal digits.

QUESTION 1(b) **[2 marks]**

(c) What would be the approximate value of the largest number expressible in IEEE single-precision floating-point (express your answer in terms of the nearest power of 2)? Suppose it was desired to represent numbers up to 256 times larger than this; how would you change the format to accommodate this? What tradeoff would be involved?

QUESTION 1(c) **[3 marks]**

Question 1 (continued)

(d) In the context of two's complement arithmetic, explain the term *sign extension*. Briefly describe one example in computer systems where sign extension is used.

QUESTION 1(d) **[2 marks]**

(e) State one similarity and one difference between registers and main memory in a computer.

QUESTION 1(e) **[1 mark]**

(f) State Moore's Law. Briefly describe its impact on computer technology over the last 50 years.

QUESTION 1(f) **[2 marks]**

Student Number: .....

QUESTION 2 [17 marks]

(a) Write C statements to declare an integer variable `i` initialized to `0`, and a char pointer `s` initialized to the string `"comp2300"`.

QUESTION 2(a)	**[1 mark]**

(b) Consider the following C program `foo.c`

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char *argv[]) {
   int a[256], i;
   int n = atoi(argv[1]);
   a[0] = 0;
   for (i=1; i<=n; i++) {
     a[i] = 2*a[i-1] + 1;
     printf("%3d", a[i]);
   }
   printf("\n");
   return 0;
}
```

Suppose the program was compiled and linked into an executable program called `foo`.

(i) Write the output produced by the command `./foo 4`, clearly indicating spaces. *Hint:* in this case, the variable `n` will be assigned the value of 4.

QUESTION 2(b)[i]	**[2 marks]**

(ii) Suppose n represents an integer and $0 \leq n < 256$. In terms of n, state what the command `./foo` n produces.

QUESTION 2(b)[ii]	**[1 mark]**

Question 2 (continued)

(iii) Describe how you would re-code the program so that the array `a` was dynamically allocated from *heap memory* (give your answer in terms of changes to the C code). For full marks, include any appropriate code for the purpose of *defensive programming*. What is the main advantage of changing the program in this fashion?

QUESTION 2(b)[iii] **[3 marks]**

(iv) State two reasons why, in general, dynamically allocated memory should be freed when the storage for that memory is no longer required.

QUESTION 2(b)[iv] **[1 mark]**

Question 2 (continued)

(c) Consider the following function definition.

```
int strcmp(const char *s1, const char *s2);
// the function compares the two strings s1 and s2.
// It returns -1, 0, or +1 if  s1 is less than, equal,
// or greater than s2, respectively.
```

For example, `strcmp("ab", "ac")`, `strcmp("ab", "ab")` and `strcmp("abc", "ab")` return `-1`, `0` and `+1`, respectively. Write an implementation of `strcmp()`. Your code must not call any other function except `strlen()`.

QUESTION 2(c) **[5 marks]**

Question 2 (continued)

(d) Given the declaration `char c, s[256];`, consider the following C code to read the next character from input into `c` and the next string from input into `s`.

```
scanf("%c", &c);
scanf("%s", s);
```

State the reason why the C address-of operator `&` is not used in the second call.

QUESTION 2(d) **[1 mark]**

(e) Consider the declarations:

```
struct timeval { /* structure to represent time */
    unsigned int tv_sec;        /* seconds */
    unsigned int tv_usec;       /* microseconds */
} t1, t2;
float t_diff;
```

and suppose that `t1` and `t2` have been initialized to contain two different times recorded on a computer. Write C code to assign to `tdiff` the difference between the time represented in `t2` and the time represented in `t1` (expressed in seconds). Note: 10^6 microseconds = 1 second.

QUESTION 2(e) **[1 mark]**

Question 2 (continued)

(f) Briefly state the purpose of a function prototype ('header') and explain how these can support separate compilation of libraries.

QUESTION 2(f) **[2 marks]**

Additional answers to QUESTION ___(___)[___]

Additional answers to QUESTION __(__)[__]

Additional answers to QUESTION __(__)[__]

Student Number: .....

Additional answers to QUESTION __(__)[__]

Additional answers to QUESTION __(__)[__]

Additional answers to QUESTION __(__)[__]

Appendix

x	2^x
-5	0.03125
-4	0.0625
-3	0.125
-2	0.25
-1	0.5
0	1
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8	256
9	512
10	1024
11	2048
12	4096
13	8192
14	16384
15	32768
16	65536

Table 1: Powers of 2 in decimal