CIS 2107
Computer Systems and Low-Level Programming
Fall 2009
Midterm

October 22, 2009

Name: ________________________________

<table>
<thead>
<tr>
<th>Page</th>
<th>Points</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>17</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>Total:</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>

**Instructions**
The exam is closed book, closed notes. You may *not* use a calculator, cell phone, etc.
For the following questions, you can assume that my home directory is the jfiore directory.

1. Unix shell stuff.

(a) If I’m in my home directory, what’s the command to compile prog.c into the executable called prog?

Solution: gcc -o prog prog.c

(b) What’s the command to run the program prog taking its input from stuff.txt and writing its output to out.txt?

Solution: prog < stuff.txt > out.txt

(c) If I’m in any directory, what’s the command to change to my home directory?

(c) cd or cd ~ or cd $HOME

(d) What’s the PATH variable?

Solution: It’s the list of directories where the shell will look for programs.

(e) If I’m in the 2107 directory, what’s the one command that I’d type in order to see the file stuff.txt one page at a time?

Solution: more ../stuff.txt or less ../stuff.txt

2. What does it mean when we write that a variable declared outside a function is static?

Solution: It means that it’s private to the file.

3. (a) 48 gbytes = ? tbits

Solution: 0.384 or 48*8/1,000

(b) 32 kbytes = ? bytes

Solution: 32,000 or 32*1,000

(c) 96 mbytes = ? kbits

Solution: 768,000 or 96*8*1,000
(1 point) (d) 32 tbytes = ? kbytes

(d) \(32,000,000,000\) or \(32 \times 10^9\)

Convert each of the following from base 10 to base 2 and base 16

(2 points) 4. (a) \(237_{10}\)

Solution:

\[
\begin{align*}
237 &= 118 \times 2 + 1 \\
118 &= 59 \times 2 + 0 \\
59 &= 29 \times 2 + 1 \\
29 &= 14 \times 2 + 1 \\
14 &= 7 \times 2 + 0 \\
7 &= 3 \times 2 + 1 \\
3 &= 1 \times 2 + 1 \\
1 &= 0 \times 2 + 1
\end{align*}
\]

\[237_{10} = 11101101_2 = ED_{16}\]

(2 points) (b) \(161_{10}\)

Solution:

\[
\begin{align*}
161 &= 80 \times 2 + 1 \\
80 &= 40 \times 2 + 0 \\
40 &= 20 \times 2 + 0 \\
20 &= 10 \times 2 + 0 \\
10 &= 5 \times 2 + 0 \\
5 &= 2 \times 2 + 1 \\
2 &= 1 \times 2 + 0 \\
1 &= 0 \times 2 + 1
\end{align*}
\]

\[161_{10} = 10100001_2 = A1_{16}\]

(2 points) (c) \(111_{10}\)

points: _______ out of a possible 9

2 of 11 exam continues...
Solution:

\[ 111 = 55 \times 2 + 1 \]
\[ 55 = 27 \times 2 + 1 \]
\[ 27 = 13 \times 2 + 1 \]
\[ 13 = 6 \times 2 + 1 \]
\[ 6 = 3 \times 2 + 0 \]
\[ 3 = 1 \times 2 + 1 \]
\[ 1 = 0 \times 2 + 1 \]

\[ 111_{10} = 1101111_2 = 6F_{16} \]

5. Using the approximation trick that we talked about in class, about how much are each of the following?

(a) \( 2^{19} \)

(b) \( 2^{23} \)

(c) \( 2^{42} \)

(a) 512 thousand

(b) 8 million

(c) 4 trillion
6. What is $11100101_2 + 10101_2$ in base 2?

Solution:

\[
\begin{array}{c}
1 & 1 & 1 & 0 & 0 & 1 & 0 & 1_2 \\
+ & 1 & 0 & 1 & 0 & 1_2 \\
\hline
1 & 1 & 1 & 1 & 1 & 0 & 1 & 0_2 \\
\end{array}
\]

which should make sense, because in decimal, we'd have:

<table>
<thead>
<tr>
<th>binary</th>
<th>decimal</th>
</tr>
</thead>
<tbody>
<tr>
<td>11100101_2</td>
<td>129_{10}</td>
</tr>
<tr>
<td>10101_2</td>
<td>21_{10}</td>
</tr>
<tr>
<td>11111010_2</td>
<td>250_{10}</td>
</tr>
</tbody>
</table>

7. What is $64A95_{16} + BDF_{16}$ in base 16?

Solution:

\[
\begin{array}{c}
1 & 1 & 1 & 1 & 1 & 1 & 4 & 16 \\
+ & 6 & 4 & A & 9 & 5 & F & 16 \\
\hline
6 & 5 & 6 & 7 & 4 & 16 \\
\end{array}
\]

8. In hex, what is the smallest integer that can be represented by a 32-bit two’s complement int?

8. \texttt{0x80000000}

9. In hex, what is the largest integer that can be represented by a 32-bit two’s complement int?

9. \texttt{0xFFFFFFFF}

10. In hex, what is $-1$ as a 32-bit two’s complement int?

10. \texttt{0xFFFFFFFF}

11. What is printed by the following?

```c
    char x=64;
    x*=2;
    printf("x=%d\n", x);
```

11. \texttt{-128}
12. **Some bit operations.** If we have `char i = 0x9C, j = 0xA3`, what is the result of the following operations? Your answer must be in the form of exactly two hex digits\(^1\).

(a) `^ i`

(b) `!i`

(c) `!!i`

(d) `i & 0xFF`

(e) `i ^ j ^ j`

(f) `i || j`

(g) `i << 2`

---

\(^1\)Yes. In real life, some operations could involve promoting the operands to 32-bit ints. Ignore that for now. Just pretend that we’re living in the land of 8-bit arithmetic.
13. **Two’s complement.** The function \( \text{TC8}() \) takes an 8-bit signed int and returns its two’s complement. What would be the value returned by applying \( \text{TC8}() \) to the following? Your answer should be in the form of two hex digits.

(a) (2 points) 0x61

(b) (2 points) 0xA0

**Solution:** The details:

(a) \(0x61 = 01100001_2\)
\(\sim 0x61 = 10011110_2\)
\(\sim 0x61 + 1 = 10011111_2\)
\(= 0x9F\)

(b) \(0xA0 = 10100000_2\)
\(\sim 0xA0 = 01011110_2\)
\(\sim 0xA0 + 1 = 01100000_2\)
\(= 0x60\)

14. If I have the following:

```c
int main(void)
{
    int a=10, b=20;
    int *p=&a;
    int *q=p;
    ...
}
```

points: ________
out of a possible 14

question 14 continues...
and memory is laid out like this:

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>1000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>b</td>
<td>1004</td>
<td></td>
<td></td>
</tr>
<tr>
<td>p</td>
<td>1008</td>
<td></td>
<td></td>
</tr>
<tr>
<td>q</td>
<td>1012</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

what do you see if you print:

1. (1 point) a
   - (a) 10
2. (1 point) &a
   - (b) 1000
3. (1 point) b
   - (c) 20
4. (1 point) &b
   - (d) 1004
5. (1 point) p
   - (e) 1000
6. (1 point) *p
   - (f) 10
7. (1 point) &p
   - (g) 1008
8. (1 point) q
   - (h) 1000
9. (1 point) *q
   - (i) 10
10. (1 point) &q
    - (j) 1012

15. What is the value of each of the following after `func()` is called?

```c
#include <stdio.h>
#include <stdlib.h>

typedef struct {
    int A[3];
    int x;
    int *p;
} Stuff;

int main(void)
{
    Stuff s;
    int x=10;
    s.p=(int*)malloc(sizeof(int));
    *(s.p)=60;
    s.A[0]=50;
    s.x=70;
    func(A, x, s);
}
```

points: ______  
out of a possible 14  
7 of 11  
question 15 continues...
(a) (2 points) \( x \)?
(a) 10

(b) (2 points) \( A[0] \)?
(b) 222

(c) (2 points) \( s.A[0] \)?
(c) 50

(d) (2 points) \( s.x \)?
(d) 70

(e) (2 points) \( *(s.p) \)?
(e) 666

(6 points) 16. Fill in the following code:

```c
#define NROWS 5
#define NCOLS 10

int *A;
int j;
int *p;

/* allocate storage for a NROWSxNCOLS array of int and let A point to it */
A=(int*)malloc(NROWS*NCOLS*sizeof(int));

/* set all items in A to 0. Do not allocate any new storage */
for (p=A, j=0; j<NROWS*NCOLS; p++, j++)
    *p=0;

/* there were a few other ways to do this */
```

(6 points) 17. Write a function which is passed an array of int and its length. The function returns the sum of the array items which are divisible by two.

**Solution:** There are several ways to do this. Here's one:
int numDivByTwo(int A[], int len)
{
    int i, count;
    for (i=0, count=0; i<len; i++)
        if (A[i]%2==0)
            count++;
    return count;
}

18. (7 points) Write a function (you may write additional helper functions if it simplifies things) which is passed a string s. The function removes all of the vowels in s. You may use functions in ctype.h, but not in string.h.

Solution:

19. Integer representation.

(a) (5 points) Write the function void printInt(int x) which prints to the screen each of the bytes of x in hex one byte per line. (Reminder: the printf format flag to print an int in hex is %x.)

Solution:

    void printInt(int x)
    {
        unsigned int i;
        char *p=(char*)&x;
        for (i=0; i<sizeof(int); i++, p++)
            printf(''%.2x
'', (unsigned char)*p);
    }

(b) (5 points) If an int 0x01234567 is passed to your function, and we’re running on a little-endian machine, what exactly would be printed to the screen? (If you didn’t do part a, you can still get full credit for this part.)

Solution: This depends on how you’ve implemented the function, but for mine, if it’s little endian, we see 0x67, 0x45, 0x23, 0x01.
(7 points) 20. Write a function which is passed a string s, and returns a string which is an acronym of s. For example, if s is “United Parcel Service”, the function returns a pointer to the string “UPS”. If s is “American Airlines”, the function returns “AA”.

Solution: Here’s one possibility:

```c
char *acro(char *s)
{
    int inword = 0, length, i, j;
    char *a;

    length=len(s);
    if ((a=malloc(length+1))==NULL)
        return NULL;
    for (i=0, j=0; i<length; i++) {
        if (inword) {
            if (isspace(s[i]))
                inword = 0;
        } else {
            if (!isspace(s[i])) {
                inword=1;
                a[j]=s[i];
                j++;
            }
        }
    } a[j]=’\0’;
    return a;
}
```

```c
int len(char *s) {
    char *e=s;
    while (*e!=’\0’) e++;
    return e-s;
}
```