Problem P1 [15 points]: Stack Conventions

Make the following assumptions:

- Assume that the size of each stack frame is exactly 20 bytes.
- Assume that the variable x in the factorial function is at a constant location in each stack frame (e.g. four bytes from the start of the stack frame).

Note:

- `printf("%x\n", x);` /* Displays the value x in hexadecimal output format (e.g. if x = 10, prints A) */
- `printf("%d\n", y);` /* Displays the value y as an integer (e.g. if x = 10, prints 10) */

```c
int fact(int n, char* p) {
    char x;
    if (p == NULL)
        p = &x;
    printf("%x\n", p - &x);
    if (n == 1)
        return 1;
    else
        return n * fact(n - 1, p);
}
```

What gets printed when the following line of code executes:

```c
int y = fact(3);
printf("%d\n", y);
```
Problem P2 [20 points]: Disassembly
Convert the following hexadecimal numbers to their MIPS instruction:

a. 0x01285022
b. 0x0211402a
b. 0x12200007
d. 0x0c100009

Problem P3 [25 points]: Control Instructions
Given the following data segment and code segment, translate the machine code to MIPS and provide the output. You only need to translate the 7 instructions below, no need to define the .data segment.

```
DATA
[0x10000000]...[0x10010000] 0x00000000
[0x10010000]          0x11d60c0c  0x0000f2f8 0x00000000  0x00000000
[0x10010010]...
CODE
0x3c111001
0x3c121007
0x82290000
0x2128002a
0xa2280000
0x22310001
0x1632ffff
```

Hint:
when the code completes, if you were to order each byte from 1001-1007 and write out their binary representation you should see a picture represented by the ones : =)
ex. Memory right now:

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1001:</td>
<td>00001100</td>
<td></td>
</tr>
<tr>
<td>1002:</td>
<td>00001100</td>
<td></td>
</tr>
<tr>
<td>1003:</td>
<td>01100000</td>
<td></td>
</tr>
</tbody>
</table>

....

clearly there is no picture at the moment.

Problem P4 [15 points]: Call/Return
Assume the following execution scenario: The content of the top of a memory stack is 5320. The content of the stack pointer SP is 3560. A two-word call subroutine instruction is located in memory at address 1120 followed by the address field of 6720 at location 1121. What are the contents of PC, SP and the top of the stack:

<table>
<thead>
<tr>
<th></th>
<th>PC</th>
<th>SP</th>
<th>TOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>After call</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>After return</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>