Solutions to End of Chapter 6 Problems

1. Given the following variable declarations:

   int foo;
   char *bar;

   and the following memory layout (all values in hexadecimal and little endian):

<table>
<thead>
<tr>
<th>Variable</th>
<th>Address</th>
<th>Hex</th>
<th>Character</th>
</tr>
</thead>
<tbody>
<tr>
<td>bar</td>
<td>080483A8</td>
<td>4F 21</td>
<td>BRAVO!</td>
</tr>
<tr>
<td>foo</td>
<td>080483AC</td>
<td>05 00</td>
<td></td>
</tr>
</tbody>
</table>

   What is the value of:
   
   (a) &foo (in hex) 0x080483AC
   (b) foo (in decimal) 5
   (c) bar (in hex) 0x080483A8
   (d) &bar (in hex) 0x080483A8
   
   (e) If we were to print out the string that bar points to, what would be printed out?

   bar is a character pointer that has a value of 0x080483A0 which points to

   The null-terminated string starting from 0x080483A0 is:

   Hex: 42 52 41 56 4F 21 00
   Character: B R A V O ! null

   BRAVO! would be printed out

2. Given the following declarations, what would be the C statement to assign ptr_age the address of the integer age? (Circle the correct answer)

   int age;
   int *ptr_age;

   (a) *ptr_age = &age;
   (b) ptr_age = age;
   (c) ptr_age = &age;
3. Given the following C snippet, what would the output of the `printf` statement be?

```c
char name[40] = "LCDR Atwood";  
char *ptr1;  
char *ptr2;

ptr1 = name;  
// ptr1 = &name[0]

ptr2 = ptr1 + 6;  
// ptr2 = &name[0] + 6 = &name[6]

strcpy(ptr2, "good day by all!");

printf("My teacher is %s\n", name);
```

```
My teacher is LCDR Agood day by all!
```
4. If you were to read the following segment of code, what would be printed to the monitor by the printf command (circle the correct answer)?

```c
float gain[6]={2.5, 7, 3, 7.8, 9};
printf("The address of the array is %x\n\n", gain);
```

(a) gain[0]  
(b) address of gain[0]  
(c) 2.5  
(d) 7  
(e) address of 7

5. From question 4, how many bytes are allocated for the array gain?

6 × 4 = 24 bytes

6. **Fill in the blank:** When calling a pointer with conversion specifiers, using %x________ will give the address the pointer contains and %s________ will give the string stored at that address.

7. What is printed out by the following code? It may be helpful to you to sketch what the memory layout would be for this problem.

```c
int k;
char a[10] = "hyperbola";
char *aptr;
aptr = a + 6;
// aptr = &a[0] + 6 = &a[6]
for (k = 0; k <= 6; k = k + 2)
    printf("%s\n",aptr-k);
```

Output:
- k = 0: ola
- k = 2: rbola
- k = 4: perbola
- k = 6: hyperbola

8. Consider the following snippet of code:

```c
int this_array[20];
int *ptr1;
ptr1 = this_array + 10;
```

If the address of this_array = 0xbffff800, then ptr1 = ___________ (in hex)?

ptr1 = this_array + 10 = 0xbffff800 + 0xA = 0xbffff80A