Solutions to Practice Problems

Practice Problem 8.1
Consider the picture above, showing program 1 in memory.
(a) How does the CPU keep track of the program's proper location within the text segment?
(b) How does the CPU keep track of where the stack is located in main memory?

Solution: (a) The eip register holds the address of the next instruction to be executed.
(b) The ebp and esp registers hold the addresses of the bottom and top of the stack.

Practice Problem 8.2
Write a snippet of C code that will allocate space on the heap for 25 characters.

Solution: 25 characters require 25 bytes. So we would use:

```c
char *ptr2;
ptr2 = (char *) malloc(25);
```

Practice Problem 8.3
Which segment of memory is physically highest (i.e., has the smallest addresses)?
(a) Heap
(b) Stack
(c) Text Segment
(d) Registers

Solution: (c)

Practice Problem 8.4
In which direction does the heap grow?
(a) From the bottom (larger memory address) up (to a smaller memory address).
(b) From the top (smaller memory address) down (to a larger memory address).
(c) It depends on the corresponding number and types of variables currently allocated on the stack.
(d) It depends on the prolonged effects of solar and liqueouscent additives combined with the chemical makeup of the heap.

Solution: (b)
Practice Problem 8.5

When we run the program above by entering:

```
./a.out 10 cyber2
```

(a) What is the value of \( \text{argc} \)?

(b) What are the values of \( \text{argv[1]} \) and \( \text{argv[2]} \)?

(b) What are the types of \( \text{argv[1]} \) and \( \text{argv[2]} \)?

Solution:

(a) 3

(b) \( \text{argv[1]} = "10" \) and \( \text{argv[2]} = "cyber2" \)

(c) Both are strings

Practice Problem 8.6

Suppose we run the program shown above with the debugger, and set a breakpoint at line 16. Which of the following is a possible value stored in the instruction pointer \( \text{eip} \)?

(a) \( 0x0804848c \)

(b) \( 0xbffff810 \)

(c) \( 0x0804a010 \)

Solution: (a)

Practice Problem 8.7

Suppose we run the program shown above with the debugger, and set a breakpoint at line 16. Which of the following is a possible address for where the variable \( \text{size} \) is stored?

(a) \( 0x0804848c \)

(b) \( 0xbffff810 \)

(c) \( 0x0804a010 \)

Solution: (b)

Practice Problem 8.8

The above picture of the stack shows that the variable \( \text{size} \) is stored "above" (i.e., at lower memory) than \( \text{ptr} \). How do we know that this must be the case?

Solution: Items are placed on the stack in the order in which they are declared.
Practice Problem 8.9

What is the output of the following program?

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

int main(int argc, char *argv[])
{
    char string1[ ] = "Happy" ;
    char string2[ ] = "Joyous" ;
    char string3[ ] = "Happy Times" ;
    char string4[ ] = "Happy" ;

    if(strcmp( string1 , string2 ) == 0)
        printf("\nString 1 and String 2 match\n");
    else
        printf("\nString 1 and String 2 do NOT match\n");

    if(strcmp( string1 , string3 ) == 0)
        printf("\nString 1 and String 3 match\n");
    else
        printf("\nString 1 and String 3 do NOT match\n");

    if(strcmp( string1 , string4 ) == 0)
        printf("\nString 1 and String 4 match\n");
    else
        printf("\nString 1 and String 4 do NOT match\n");
}
```

Solution:

```
midshipman@EC310:~/work $ ./a.out
String 1 and String 2 do NOT match
String 1 and String 3 do NOT match
String 1 and String 4 match
midshipman@EC310:~/work $ 
```