Solutions to Practice Problems

Practice Problem 5.1

In the screen capture above, at what assembly language instruction did the program stop? That is, what is the next instruction that will execute, and what is the address in main memory where this instruction is stored?

Solution:

Next instruction: \texttt{mov DWORD PTR [ebp-4],0x7}
Address: \texttt{0x08048354}

Practice Problem 5.2

What is the value (address) stored in the \texttt{eip} register?

Solution: \texttt{0x8048354}

Practice Problem 5.3

Refer to the pictures shown above. What should be printed out by each of the following commands? In each case, enter the command to confirm your answer.

Solution:

\begin{itemize}
  \item[(a)] \texttt{x/xb 0x08048354} \hspace{1em} \texttt{0xc7}
  \item[(b)] \texttt{x/xb 0x08048355} \hspace{1em} \texttt{0x45}
  \item[(c)] \texttt{x/xb 0x08048356} \hspace{1em} \texttt{0xfc}
  \item[(d)] \texttt{x/xb 0x08048357} \hspace{1em} \texttt{0x07}
\end{itemize}

Practice Problem 5.4

What will be displayed by the command: \texttt{x/xh 0x08048354}? What do you notice about the order of the bytes in the value displayed?

Solution:

\texttt{x/xh 0x08048354} \hspace{1em} \texttt{0x45c7}

notice that it reverses the little-endian order

Practice Problem 5.5

What do you think will be displayed by the command: \texttt{x/xw 0x08048354}? Confirm your result.

Solution:

\texttt{x/xw 0x08048354} \hspace{1em} \texttt{0x07fc45c7}
Practice Problem 5.6
What do you think will be displayed by the command: x/4xb 0x08048354? Confirm your result.

Solution:

    x/4b 0x08048354 0xc7 0x45 0xfc 0x07

Practice Problem 5.7
What do you think will be displayed by the command: x/xb $eip? Confirm your result.

Solution:

    x/xb $eip 0xc7

Practice Problem 5.8
What do you think will be displayed by the command: x/2xb $eip? Confirm your result.

Solution:

    x/xh $eip 0xc7 0x45

Practice Problem 5.9
What do you think will be displayed by the command: x/i $eip? Confirm your result.

Solution:

    x/i $eip mov DWORD PTR [ebp-4],0x7

Practice Problem 5.10
When you execute a command (as you just did when you typed nexti), what happens to the instruction pointer (eip)?

Solution: eip will be incremented to hold the address of the next (machine language) instruction

Practice Problem 5.11
What is the value (address) stored in the eip register? Does this answer make sense?

Solution: i r eip 0x804835b
**Practice Problem 5.12**

The instruction corresponding to `int x = 7;` has just been executed. What should I type to examine memory to see the integer value 7 on the stack? Confirm your result!

Solution:  \texttt{x/xw \$ebp-4}  

0xbffff814: \ 0x00000007

**Practice Problem 5.13**

What assembly language instruction is located at 0x0804835b?

Solution:  \texttt{x/i 0x804835b mov DWORD PTR [ebp-4], 0x7d1}

**Practice Problem 5.14**

Sketch what you expect the stack to look like after the instruction at address 0x0804835b is executed. Fill in the blocks in the memory diagram.

Solution:

```
<table>
<thead>
<tr>
<th>Address</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>bffff810</td>
<td></td>
</tr>
<tr>
<td>bffff811</td>
<td></td>
</tr>
<tr>
<td>bffff812</td>
<td></td>
</tr>
<tr>
<td>bffff813</td>
<td></td>
</tr>
<tr>
<td>bffff814</td>
<td>d1</td>
</tr>
<tr>
<td>bffff815</td>
<td>07</td>
</tr>
<tr>
<td>bffff816</td>
<td>00</td>
</tr>
<tr>
<td>bffff817</td>
<td>00</td>
</tr>
<tr>
<td>ebp</td>
<td>bffff818</td>
</tr>
</tbody>
</table>
```

**Practice Problem 5.15**

What two things happened when you entered \texttt{nexti}?  

Solution: 1. The instruction at address held in eip is executed  
2. The eip register is updated to hold address of next instruction

**Practice Problem 5.16**

What should you type to examine memory for the hex values you sketched in Practice Problem 5.14? Confirm your result!

Solution:  \texttt{x/4xb \$ebp-4}  

0xbffff814: \ 0xd1 0x07 0x00 0x00
Practice Problem 5.17

What should you type to examine memory where the integer value 2001 is stored? Confirm your result!

This question is somewhat vague, so any of the following answers would be acceptable:

Solution 1: \texttt{x/4xb \$ebp-4}

\begin{verbatim}
0xbffff814: 0xd1 0x07 0x00 0x00
\end{verbatim}

Solution 2: \texttt{x/4xb 0xbffff814}

\begin{verbatim}
0xbffff814: 0xd1 0x07 0x00 0x00
\end{verbatim}

Solution 3: \texttt{x/xw \$ebp-4}

\begin{verbatim}
0xbffff814: 0x000007d1
\end{verbatim}

Solution 4: \texttt{x/xw 0xbffff814}

\begin{verbatim}
0xbffff814: 0x000007d1
\end{verbatim}

Solution 5: \texttt{x/u \$ebp-4}

\begin{verbatim}
0xbffff814: 2001
\end{verbatim}

Solution 6: \texttt{x/u 0xbffff814}

\begin{verbatim}
0xbffff814: 2001
\end{verbatim}