ake Test: Quiz 5

Top of Form

[**[](https://webcourses.niu.edu/webapps/assessment/take/launch.jsp?course_assessment_id=_172277_1&course_id=_304209_1&content_id=_5908149_1&step=null)Test Information**](https://webcourses.niu.edu/webapps/assessment/take/launch.jsp?course_assessment_id=_172277_1&course_id=_304209_1&content_id=_5908149_1&step=null)

|  |  |
| --- | --- |
| Description | Microcode |
| Instructions |  |
| Multiple Attempts | Not allowed. This test can only be taken once. |
| Force Completion | This test can be saved and resumed later. |

Expand Question Completion Status:

**QUESTION 1**

1. How many memory busses does a Harvard machine have?

|  |  |  |
| --- | --- | --- |
|  |  | one |
|  |  | two |
|  |  | three |
|  |  | four |

**1 points**

**QUESTION 2**

1. How many memory busses does a Von Neumann machine have?

|  |  |  |
| --- | --- | --- |
|  |  | one |
|  |  | two |
|  |  | three |
|  |  | four |

**1 points**

**QUESTION 3**

1. Which of the following pseudocode describes a Predecrement PUSH instruction?

See section 2.6.1 of the "M68000 Family Programmer's Reference Manual" and/or search for discussions such as: <https://en.wikipedia.org/wiki/Stack_register> for details on how a stack works.

|  |  |  |
| --- | --- | --- |
|  |  | MAR <- Rb         MBR IN <- mem         Ra <- MBR\_IN         Rb <- Rb + 1 |
|  |  | Rb <- Rb - 1         MAR <- Rb         MBR\_IN <- mem         Ra <- MBR\_IN |
|  |  | MAR <- Rb         MBR\_OUT <- Ra         mem(MAR) <- MBR\_OUT         Rb <- Rb + 1 |
|  |  | Rb <- Rb - 1         MAR <- Rb         MBR\_OUT <- Ra         mem(MAR) <- MBR\_OUT |

**1 points**

**QUESTION 4**

1. Which of the following pseudocode describes a Postincrement POP instruction?

See section 2.6.1 of the "M68000 Family Programmer's Reference Manual" and/or search for discussions such as: <https://en.wikipedia.org/wiki/Stack_register> for details on how a stack works.

|  |  |  |
| --- | --- | --- |
|  |  | Rb <- Rb - 1         MAR <- Rb         MBR\_OUT <- Ra         mem(MAR) <- MBR\_OUT |
|  |  | MAR <- Rb         MBR IN <- mem         Ra <- MBR\_IN         Rb <- Rb + 1 |
|  |  | Rb <- Rb - 1         MAR <- Rb         MBR\_IN <- mem         Ra <- MBR\_IN |
|  |  | MAR <- Rb         MBR\_OUT <- Ra         mem(MAR) <- MBR\_OUT         Rb <- Rb + 1 |

**1 points**

**QUESTION 5**

1. With the same machine used in assignment 3, which of the following microcode fragments will perform an opcode fetch?

|  |  |  |
| --- | --- | --- |
|  |  | 0001 0  0 0 0  0 1 0 0 0  0 0 7  0 4 0 7 0002         0002 0  0 0 0  0 0 0 0 1  0 0 7  0 4 0 7 0003         0003 0  0 0 0  0 0 0 0 0  0 0 7  0 4 0 7 0004          0004 0  0 0 1  0 0 0 0 0  1 0 4  0 4 0 7 0005         0005 0  0 0 1  0 0 0 0 0  0 0 4  0 4 0 7 0006 |
|  |  | 0001 0  0 0 0  0 1 0 0 0  0 0 7  0 4 0 7 0002          0002 0  0 0 0  0 0 0 0 0  0 0 7  0 4 0 7 0003          0003 0  0 0 0  0 0 0 0 1  0 0 7  0 7 0 7 0004          0004 0  0 0 0  0 0 0 0 0  0 0 7  0 7 0 7 0005          0005 0  0 0 0  0 1 0 0 0  0 0 7  0 7 0 4 0006          0006 0  0 0 0  0 0 0 0 0  0 0 7  0 7 0 4 0007 |
|  |  | 0001 0  0 0 0  0 1 0 0 0  0 0 5  0 4 0 7 0001         0002 0  0 0 0  0 0 0 0 0  0 0 5  0 4 0 7 0001         0003 0  0 0 0  0 0 0 0 1  0 0 5  0 4 0 7 0001         0004 0  0 0 0  0 0 0 0 0  0 0 5  0 4 0 7 0001         0005 0  0 0 0  0 0 0 0 0  1 0 5  0 7 0 4 0001         0006 0  0 0 0  0 0 0 0 0  0 0 5  0 7 0 4 0001 |
|  |  | 0001 0  0 0 0  0 1 0 0 0  0 0 5  0 4 0 7 0002         0002 0  0 0 0  0 0 0 0 0  0 0 5  0 4 0 7 0003         0003 0  0 0 0  0 0 0 0 1  0 0 5  0 4 0 7 0004         0004 0  0 0 0  0 0 0 0 0  0 0 5  0 4 0 7 0005         0005 0  0 0 0  0 0 0 0 0  1 0 5  0 7 0 4 0006         0006 0  0 0 0  0 0 0 0 0  0 0 5  0 7 0 4 0007 |

**1 points**

**QUESTION 6**

1. With the same machine used in assignment 3, which of the following will add 1 to the PC register?

|  |  |  |
| --- | --- | --- |
|  |  | 0001 0  0 1 1  0 0 0 0 0  1 0 6  1 4 0 7 0002         0002 0  0 1 1  0 0 0 0 0  0 0 4  0 4 0 7 0003 |
|  |  | 0001 0  1 0 0  0 0 1 0 0  1 0 4  0 4 0 0 0002         0002 0  1 0 0  0 0 0 0 0  0 0 4  0 4 0 0 0003 |
|  |  | 0001 0  0 0 1  0 0 0 0 0  1 0 4  0 4 0 7 0002         0002 0  0 0 1  0 0 0 0 0  0 0 4  0 4 0 7 0003 |
|  |  | 0001 0  0 0 1  0 0 0 0 0  1 0 5  0 7 0 7 0002         0002 0  0 0 1  0 0 0 0 0  0 0 5  0 7 0 7 0003 |

**1 points**

**QUESTION 7**

1. With the same machine used in assignment 3, which of the following will use the current value of the flags register to determine the next microcode instruction to execute?

|  |  |  |
| --- | --- | --- |
|  |  | 1d00 0  0 0 0  0 0 0 0 0  0 0 0  0 7 0 7 1d10 |
|  |  | 1d00 1  0 0 0  0 0 0 0 0  0 0 0  0 7 0 7 1d10 |
|  |  | 1d00 2  0 0 0  0 0 0 0 0  0 0 0  0 7 0 7 1d10 |
|  |  | 1d00 3  0 0 0  0 0 0 0 0  0 0 0  0 7 0 7 1d10 |

**1 points**

**QUESTION 8**

1. With the same machine used in assignment 3, which of the following will use the current value of the IR register to determine the next microcode instruction to execute?

|  |  |  |
| --- | --- | --- |
|  |  | 1d00 0  0 0 0  0 0 0 0 0  0 0 0  0 7 0 7 1d10 |
|  |  | 1d00 1  0 0 0  0 0 0 0 0  0 0 0  0 7 0 7 1d10 |
|  |  | 1d00 2  0 0 0  0 0 0 0 0  0 0 0  0 7 0 7 1d10 |
|  |  | 1d00 3  0 0 0  0 0 0 0 0  0 0 0  0 7 0 7 1d10 |

**1 points**

**QUESTION 9**

1. What is stored in the 'text' memory region of a program running on a PC?

|  |  |  |
| --- | --- | --- |
|  |  | The words typed into a text editor. |
|  |  | The stack data. |
|  |  | Any iInitialized data. |
|  |  | Executable code. |

**1 points**

**QUESTION 10**

1. In the following program, where will the data for the variable y be stored?

int x;  
int q=123;  
  
int main()  
{  
    int y;   
    static int a;  
    static int b = 23;  
  
    return;  
}

|  |  |  |
| --- | --- | --- |
|  |  | text |
|  |  | bss |
|  |  | heap |
|  |  | stack |

Bottom of Form