Assembly Language Worksheet

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Part I: Fetch-Execute Cycle**

|  |  |  |  |
| --- | --- | --- | --- |
| **Registers** | | **Memory Address** | **Instruction or Data** |
| **PC** | 5 | #100 | 5 |
| **R** | 5 | #101 | 3 |
| **IR** | LOAD #100 | #102 | 1 |
|  |  | #103 | 5 |
|  |  |  |  |

1. Answer the questions in this section starting with the values above. Assume we are stepping through the instructions of one program so the changes made are cumulative.
   1. Fill in the values of the memory cells after the following command is executed: LOAD #101

|  |  |  |  |
| --- | --- | --- | --- |
| **Registers** | | **Memory Address** | **Instruction or Data** |
| **PC** | 6 | #100 |  |
| **R** |  | #101 |  |
| **IR** | LOAD #101 | #102 |  |
|  |  | #103 |  |
|  |  |  |  |

* 1. Fill in the values of the memory cells after the following command is executed: ADD #102

|  |  |  |  |
| --- | --- | --- | --- |
| **Registers** | | **Memory Address** | **Instruction or Data** |
| **PC** | 7 | #100 |  |
| **R** |  | #101 |  |
| **IR** | ADD #102 | #102 |  |
|  |  | #103 |  |
|  |  |  |  |

* 1. Fill in the values of the memory cells after the following command is executed: SUBTRACT #100

|  |  |  |  |
| --- | --- | --- | --- |
| **Registers** | | **Memory Address** | **Instruction or Data** |
| **PC** | 8 | #100 |  |
| **R** |  | #101 |  |
| **IR** | SUBTRACT #100 | #102 |  |
|  |  | #103 |  |
|  |  |  |  |

* 1. Fill in the values of the memory cells after the following command is executed: STORE #100

|  |  |  |  |
| --- | --- | --- | --- |
| **Registers** | | **Memory Address** | **Instruction or Data** |
| **PC** | 9 | #100 |  |
| **R** |  | #101 |  |
| **IR** | STORE #100 | #102 |  |
|  |  | #103 |  |
|  |  |  |  |

**Part II: Python Code to Assembly Language**

1. Convert the following Python code into assembly language code. Start your code at memory cell 0 and assume that variables a, b, c, & d are stored in memory cells 101, 102, 103, & 104.

d = a + b - c

if a > b:

c = a

1. Convert the following Python code into assembly language code. Start your code at memory cell 0 and assume that variables a, b, c, & d are stored in memory cells 101, 102, 103, & 104 *(Hint: Your answer from the question above may be a good start.)*

d = a + b - c

if a > b:

c = a

if a == b:

a = b - c

1. Convert the following Python code into assembly language code. Start your code at memory cell 0 and assume that variables a, b, c, & d are stored in memory cells 101, 102, 103, & 104 *(Hint: Your answer from the question above may be a good start.)*

d = a + b - c

if a > b:

c = a

elif a == b:

a = b - c

else:

a = a - 1

**Part III: Assembly Language to Python Code**

1. Convert the following assembly language code into Python code; assume that variables A, B, and C are stored in memory cells 101, 102, and 103, respectively.

|  |  |
| --- | --- |
| Address | Instruction |
| 1 | LOAD C |
| 2 | SUBTRACT A |
| 3 | ADD B |
| 4 | ADD B |
| 5 | STORE C |
| 6 | INCREMENT A |
| 7 | HALT |

1. Convert the following assembly language code into Python code. In your answer, the variables stored at locations 101, 102, & 103 should be named a, b, & c.

|  |  |  |  |
| --- | --- | --- | --- |
| **Registers** | | **Memory Address** | **Instruction or Data** |
| **PC** |  | #1 | LOAD 102 |
| **R** |  | #2 | COMPARE 101 |
| **IR** |  | #3 | JUMPLT 9 |
| **CCR** |  | #4 | JUMPEQ 9 |
|  |  | #5 | LOAD 103 |
|  |  | #6 | SUBSTRACT 101 |
|  |  | #7 | STORE 103 |
|  |  | #8 | JUMP 11 |
|  |  | #9 | LOAD 101 |
|  |  | #10 | STORE 102 |
|  |  | #11 | DECREMENT 101 |
|  |  | #12 | HALT |
|  |  | . |  |
|  |  | . |  |
|  |  | #100 |  |
|  |  | #101 |  |
|  |  | #102 |  |
|  |  | #103 |  |