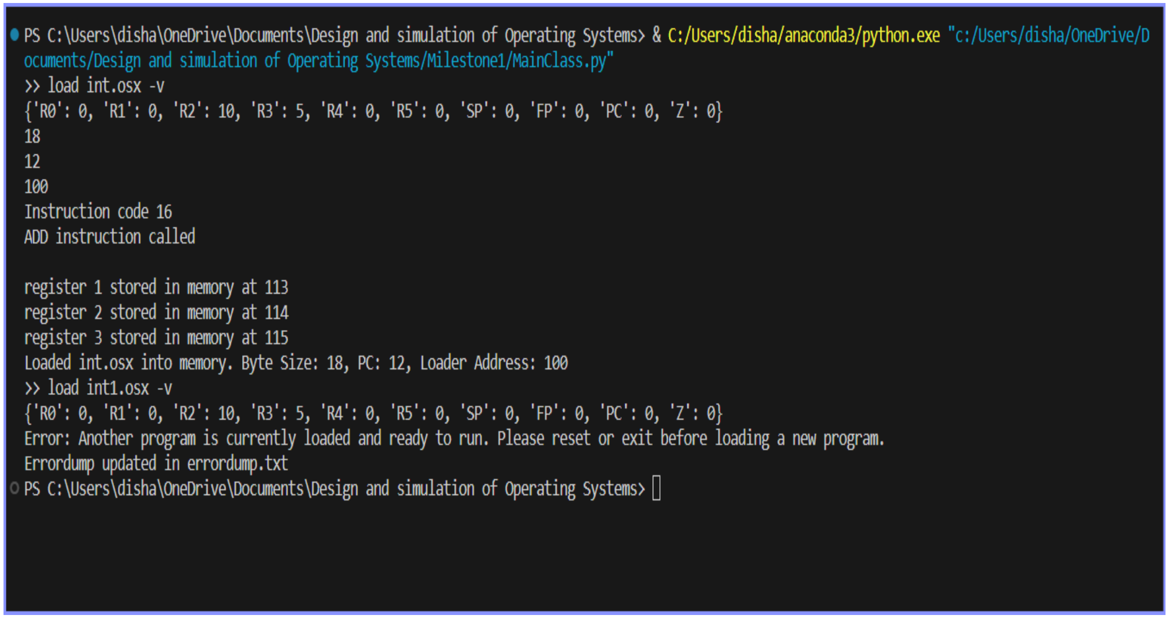
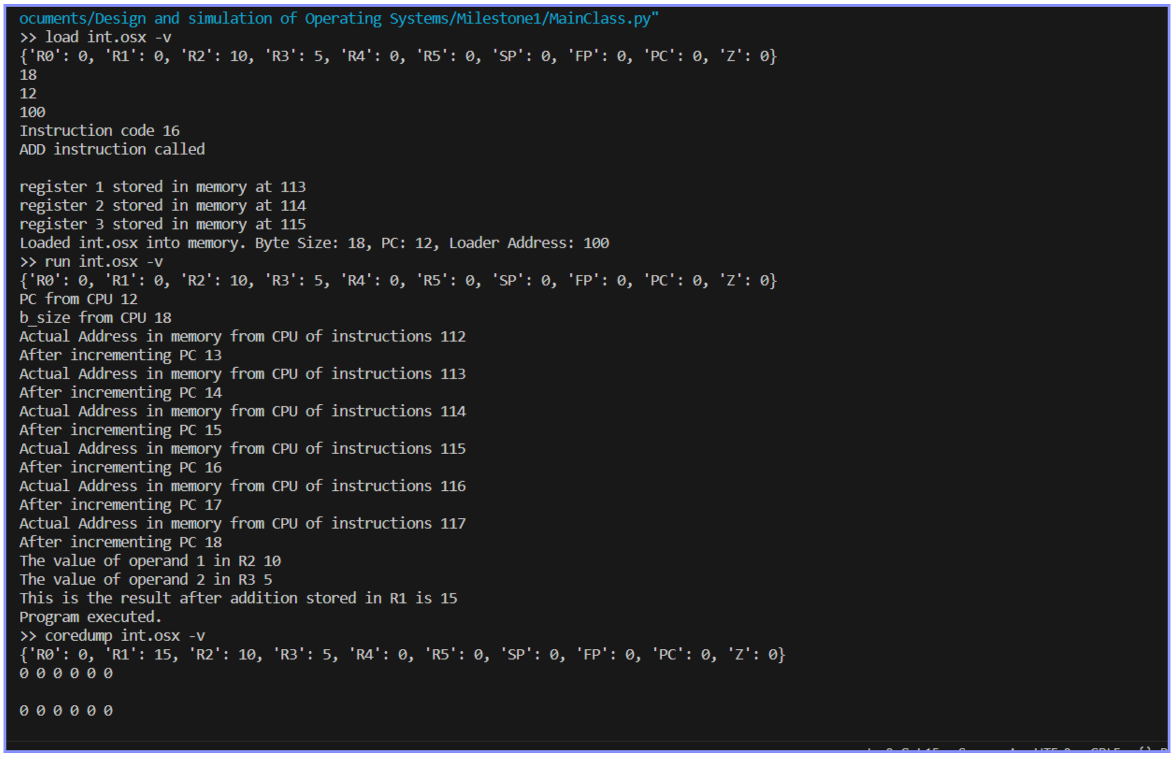
Test Cases for Milestone 1

1. Program Load Exception while other program is already loaded in the memory.

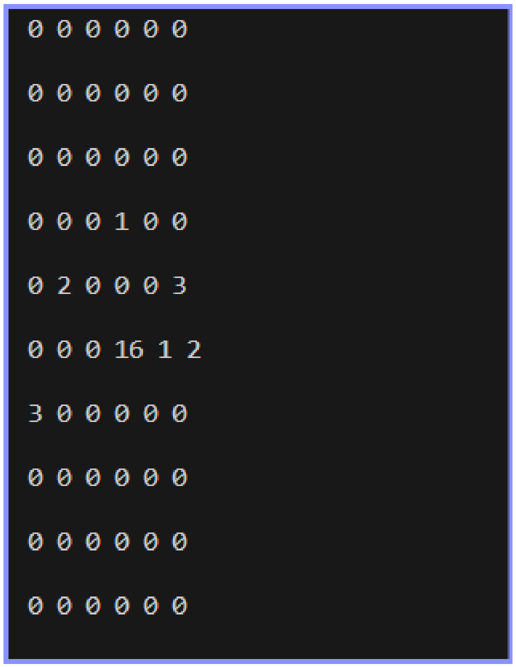


```
PS C:\Users\disha\OneDrive\Documents\Design and simulation of Operating Systems> & C:/Users/disha/anaconda3/python.exe "c:/Users/disha/OneDrive/Documents/Design and simulation of Operating Systems/Milestone1/MainClass.py"
>> load int.osx -v
{'R0': 0, 'R1': 0, 'R2': 10, 'R3': 5, 'R4': 0, 'R5': 0, 'SP': 0, 'FP': 0, 'PC': 0, 'Z': 0}
18
12
100
Instruction code 16
ADD instruction called

register 1 stored in memory at 113
register 2 stored in memory at 114
register 3 stored in memory at 115
Loaded int.osx into memory. Byte Size: 18, PC: 12, Loader Address: 100
>> load int1.osx -v
{'R0': 0, 'R1': 0, 'R2': 10, 'R3': 5, 'R4': 0, 'R5': 0, 'SP': 0, 'FP': 0, 'PC': 0, 'Z': 0}
Error: Another program is currently loaded and ready to run. Please reset or exit before loading a new program.
Errordump updated in errordump.txt
PS C:\Users\disha\OneDrive\Documents\Design and simulation of Operating Systems>
```

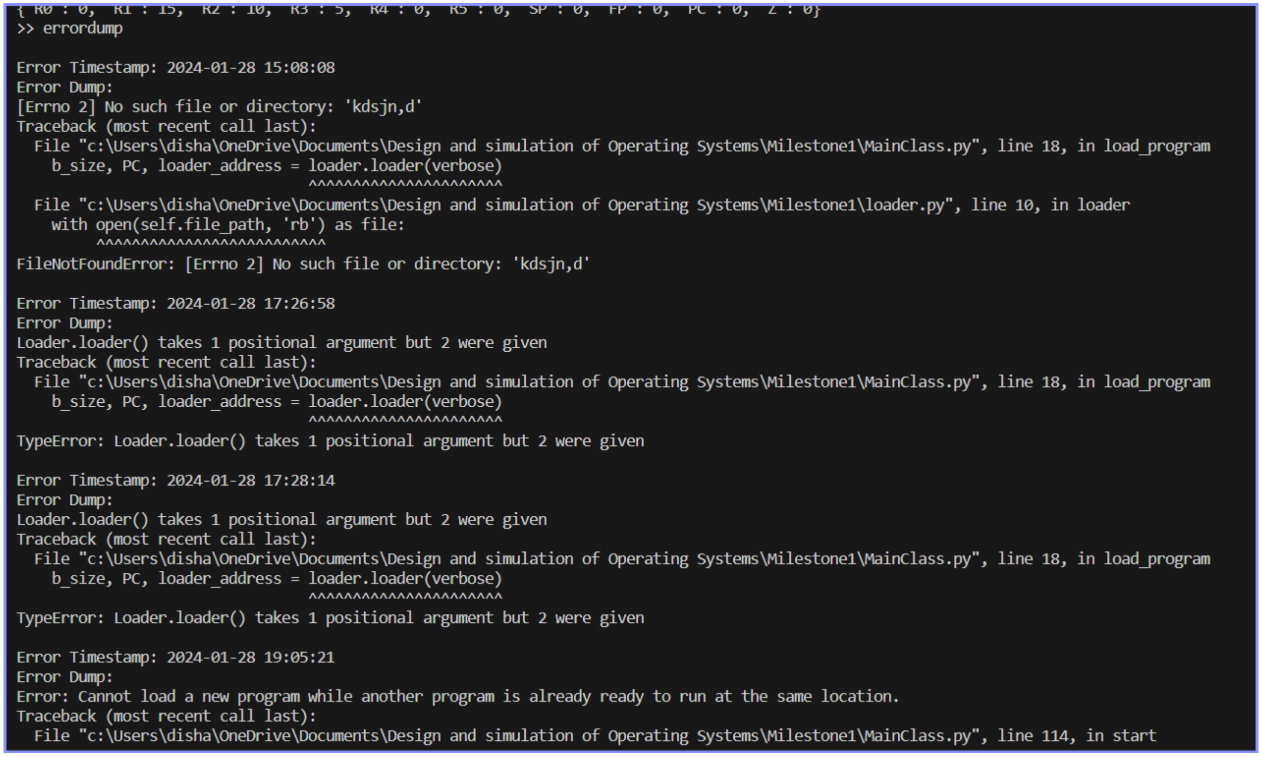
2. Test Case 1 file:
.WORD 1
.WORD 2
.WORD 3
ADD R1 R2 R3



```
ocuments/Design and simulation of Operating Systems/Milestone1/MainClass.py"
>> load int.osx -v
{'R0': 0, 'R1': 0, 'R2': 10, 'R3': 5, 'R4': 0, 'R5': 0, 'SP': 0, 'FP': 0, 'PC': 0, 'Z': 0}
18
12
100
Instruction code 16
ADD instruction called

register 1 stored in memory at 113
register 2 stored in memory at 114
register 3 stored in memory at 115
Loaded int.osx into memory. Byte Size: 18, PC: 12, Loader Address: 100
>> run int.osx -v
{'R0': 0, 'R1': 0, 'R2': 10, 'R3': 5, 'R4': 0, 'R5': 0, 'SP': 0, 'FP': 0, 'PC': 0, 'Z': 0}
PC from CPU 12
b_size from CPU 18
Actual Address in memory from CPU of instructions 112
After incrementing PC 13
Actual Address in memory from CPU of instructions 113
After incrementing PC 14
Actual Address in memory from CPU of instructions 114
After incrementing PC 15
Actual Address in memory from CPU of instructions 115
After incrementing PC 16
Actual Address in memory from CPU of instructions 116
After incrementing PC 17
Actual Address in memory from CPU of instructions 117
After incrementing PC 18
The value of operand 1 in R2 10
The value of operand 2 in R3 5
This is the result after addition stored in R1 is 15
Program executed.
>> coredump int.osx -v
{'R0': 0, 'R1': 15, 'R2': 10, 'R3': 5, 'R4': 0, 'R5': 0, 'SP': 0, 'FP': 0, 'PC': 0, 'Z': 0}
0 0 0 0 0 0

0 0 0 0 0 0
```



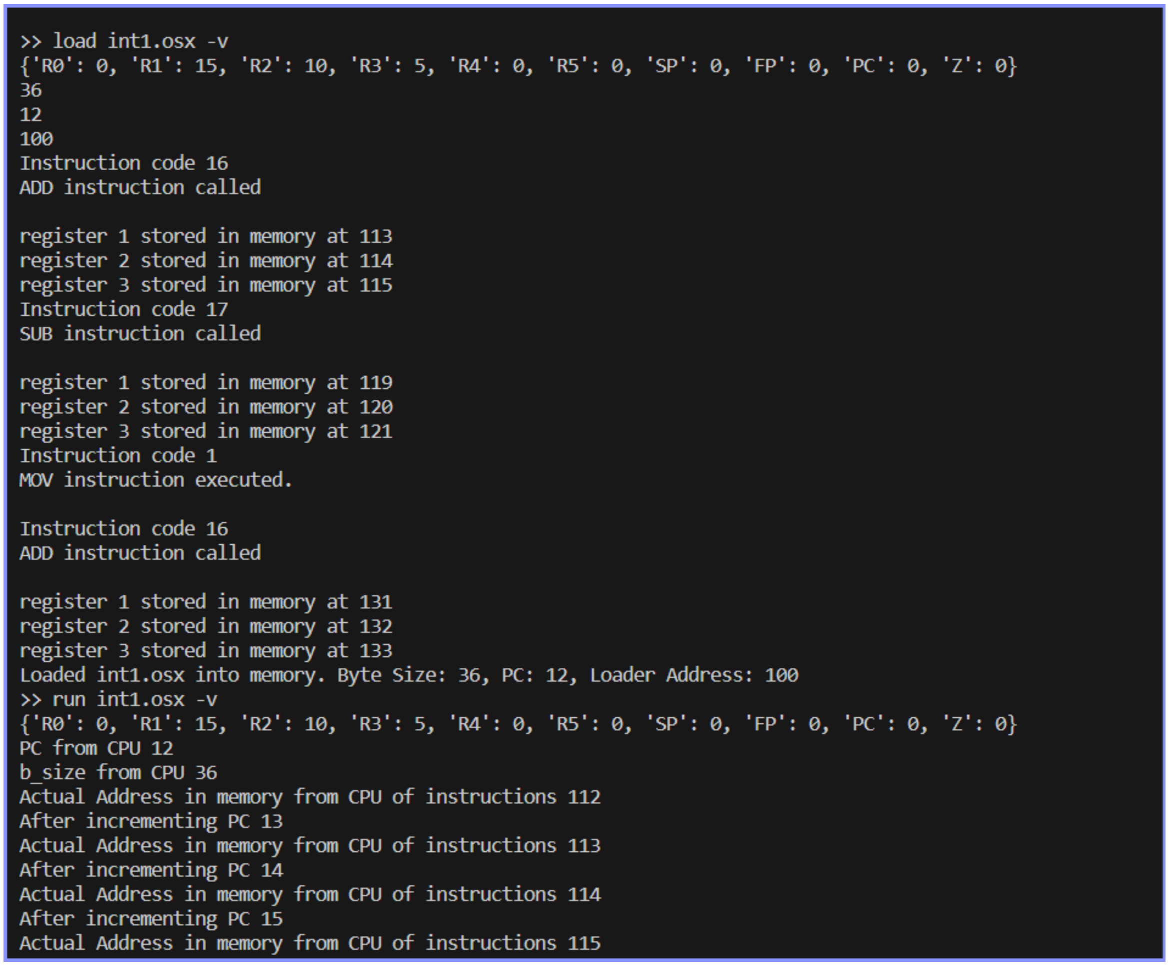
```
0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 1 0 0

0 2 0 0 0 3

0 0 0 16 1 2

3 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0
```



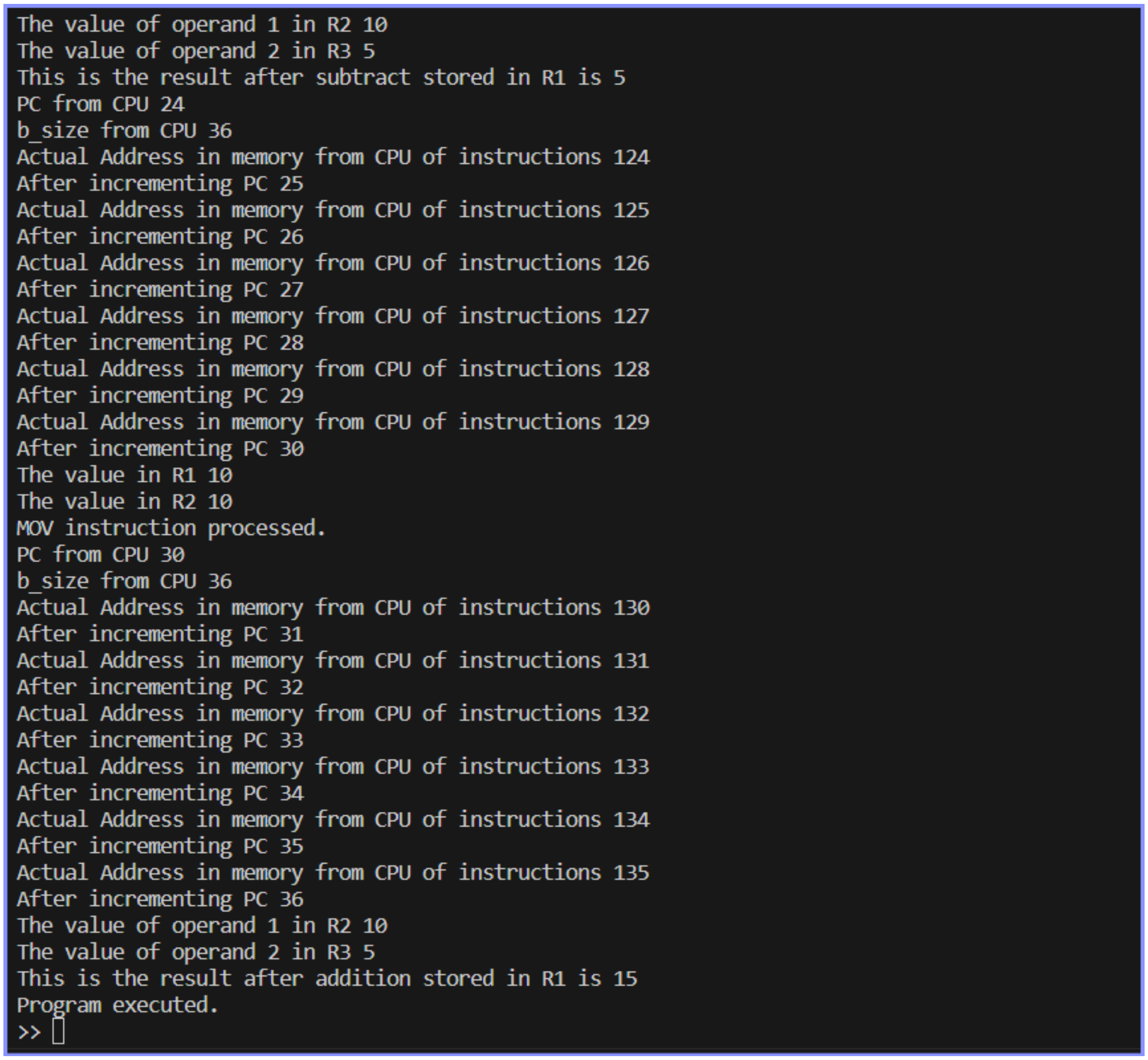
```
{ R0 : 0, R1 : 15, R2 : 10, R3 : 5, R4 : 0, R5 : 0, SP : 0, FP : 0, PC : 0, Z : 0}
>> errordump

Error Timestamp: 2024-01-28 15:08:08
Error Dump:
[Errno 2] No such file or directory: 'kdsjn,d'
Traceback (most recent call last):
  File "c:\Users\disha\OneDrive\Documents\Design and simulation of Operating Systems\Milestone1\MainClass.py", line 18, in load_program
    b_size, PC, loader_address = loader.loader(verbose)
                                 ^^^^^^^^^^^^^^^^^^^^^^
  File "c:\Users\disha\OneDrive\Documents\Design and simulation of Operating Systems\Milestone1\loader.py", line 10, in loader
    with open(self.file_path, 'rb') as file:
         ^^^^^^^^^^^^^^^^^^^^^^^^^^
FileNotFoundError: [Errno 2] No such file or directory: 'kdsjn,d'

Error Timestamp: 2024-01-28 17:26:58
Error Dump:
Loader.loader() takes 1 positional argument but 2 were given
Traceback (most recent call last):
  File "c:\Users\disha\OneDrive\Documents\Design and simulation of Operating Systems\Milestone1\MainClass.py", line 18, in load_program
    b_size, PC, loader_address = loader.loader(verbose)
                                 ^^^^^^^^^^^^^^^^^^^^^^
TypeError: Loader.loader() takes 1 positional argument but 2 were given

Error Timestamp: 2024-01-28 17:28:14
Error Dump:
Loader.loader() takes 1 positional argument but 2 were given
Traceback (most recent call last):
  File "c:\Users\disha\OneDrive\Documents\Design and simulation of Operating Systems\Milestone1\MainClass.py", line 18, in load_program
    b_size, PC, loader_address = loader.loader(verbose)
                                 ^^^^^^^^^^^^^^^^^^^^^^
TypeError: Loader.loader() takes 1 positional argument but 2 were given

Error Timestamp: 2024-01-28 19:05:21
Error Dump:
Error: Cannot load a new program while another program is already ready to run at the same location.
Traceback (most recent call last):
  File "c:\Users\disha\OneDrive\Documents\Design and simulation of Operating Systems\Milestone1\MainClass.py", line 114, in start
```

3. Test Case 2 file:
   .WORD 1
   .WORD 2
   .WORD 3
   ADD R1 R2 R3

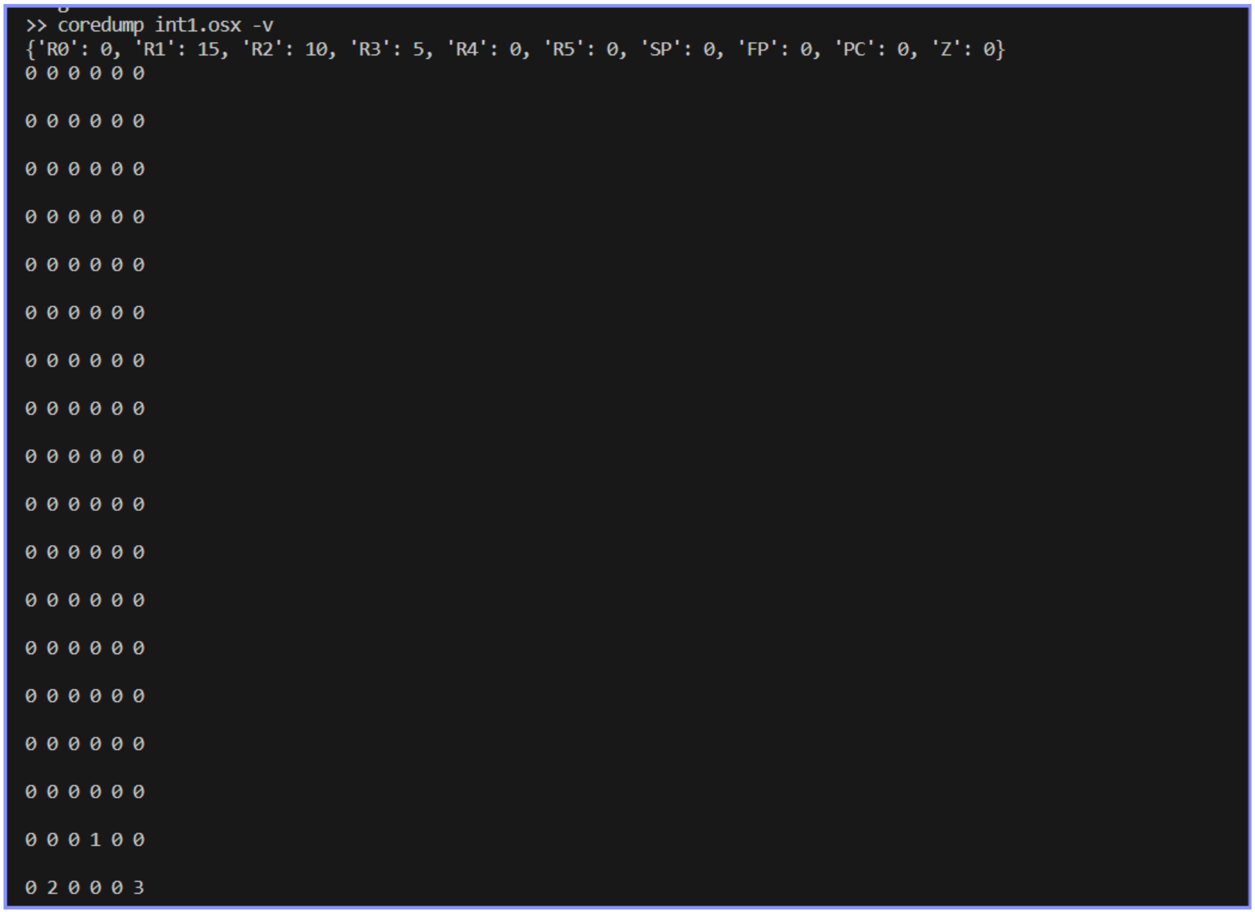
SUB R1 R2 R3
MOV R1 R2
ADD R1 R2 R3



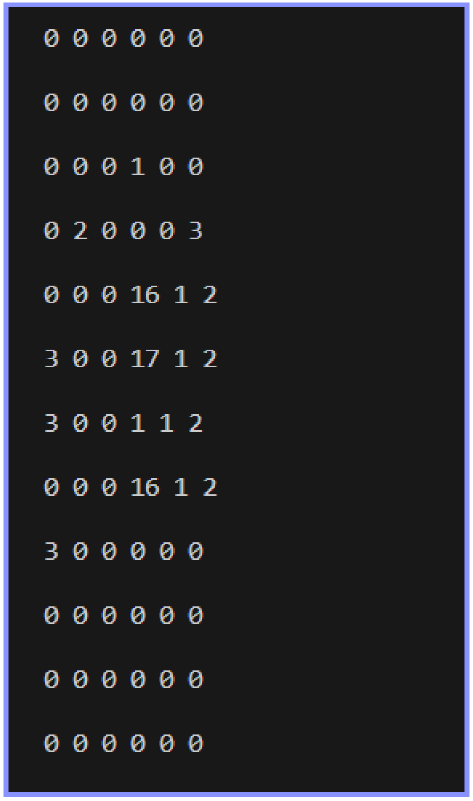
```
>> load int1.osx -v
{'R0': 0, 'R1': 15, 'R2': 10, 'R3': 5, 'R4': 0, 'R5': 0, 'SP': 0, 'FP': 0, 'PC': 0, 'Z': 0}
36
12
100
Instruction code 16
ADD instruction called

register 1 stored in memory at 113
register 2 stored in memory at 114
register 3 stored in memory at 115
Instruction code 17
SUB instruction called

register 1 stored in memory at 119
register 2 stored in memory at 120
register 3 stored in memory at 121
Instruction code 1
MOV instruction executed.

Instruction code 16
ADD instruction called

register 1 stored in memory at 131
register 2 stored in memory at 132
register 3 stored in memory at 133
Loaded int1.osx into memory. Byte Size: 36, PC: 12, Loader Address: 100
>> run int1.osx -v
{'R0': 0, 'R1': 15, 'R2': 10, 'R3': 5, 'R4': 0, 'R5': 0, 'SP': 0, 'FP': 0, 'PC': 0, 'Z': 0}
PC from CPU 12
b_size from CPU 36
Actual Address in memory from CPU of instructions 112
After incrementing PC 13
Actual Address in memory from CPU of instructions 113
After incrementing PC 14
Actual Address in memory from CPU of instructions 114
After incrementing PC 15
Actual Address in memory from CPU of instructions 115
```



```
The value of operand 1 in R2 10
The value of operand 2 in R3 5
This is the result after subtract stored in R1 is 5
PC from CPU 24
b_size from CPU 36
Actual Address in memory from CPU of instructions 124
After incrementing PC 25
Actual Address in memory from CPU of instructions 125
After incrementing PC 26
Actual Address in memory from CPU of instructions 126
After incrementing PC 27
Actual Address in memory from CPU of instructions 127
After incrementing PC 28
Actual Address in memory from CPU of instructions 128
After incrementing PC 29
Actual Address in memory from CPU of instructions 129
After incrementing PC 30
The value in R1 10
The value in R2 10
MOV instruction processed.
PC from CPU 30
b_size from CPU 36
Actual Address in memory from CPU of instructions 130
After incrementing PC 31
Actual Address in memory from CPU of instructions 131
After incrementing PC 32
Actual Address in memory from CPU of instructions 132
After incrementing PC 33
Actual Address in memory from CPU of instructions 133
After incrementing PC 34
Actual Address in memory from CPU of instructions 134
After incrementing PC 35
Actual Address in memory from CPU of instructions 135
After incrementing PC 36
The value of operand 1 in R2 10
The value of operand 2 in R3 5
This is the result after addition stored in R1 is 15
Program executed.
>>
```



```
>> coredump int1.osx -v
{'R0': 0, 'R1': 15, 'R2': 10, 'R3': 5, 'R4': 0, 'R5': 0, 'SP': 0, 'FP': 0, 'PC': 0, 'Z': 0}
0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 1 0 0

0 2 0 0 0 3
```



```
0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 1 0 0

0 2 0 0 0 3

0 0 0 16 1 2

3 0 0 17 1 2

3 0 0 1 1 2

0 0 0 16 1 2

3 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0

0 0 0 0 0 0
```