

Part 2 (run >make all, then >./A)

- A(), B(), and C() are in 3 different files.
- main() calls A().
- A() creates a 64-bit integer and passes it to B().
- Within A() expected 8-byte ASCII string is also generated for grader reference.
- B() uses write() system call to print value stored in rsp.
- The last 8 characters on screen represent the interpreted 8-byte ASCII string.
- The address of C() is pushed to the stack and ret instruction is called.
- Control is passed to C().
- C() terminates the program with exit() system call.

Note: File A.c has 2 sample 64-bit integer numbers

```
vishwesh:Assignment 1$ cd Part\ 2
vishwesh:Part 2$ make all
nasm -f elf64 B.asm -o B.o
gcc A.c B.o -o A -no-pie
vishwesh:Part 2$ ./A
-----
Entered A
-----
64 Bit Number: 4702111234474983745
Expected String: AAAAAAAA
-----
Entered B
-----
^@OAAAAAAAA
-----
Entered C
-----
```