

**Question 5.3–1:** (Solution, p 2) Translate the following assembly language program into machine language. Express your answer in bits.

	addr	data
	00000	
READ	00001	
top: WRITE	00010	
ADD one	00011	
JPOS top	00100	
HALT	00101	
one: 1	00110	
	00111	

**Question 5.3–2:** (Solution, p 2) Write an assembly language program that reads a number  $n$  and displays the value  $4n + 3$ .

**Question 5.3–3:** (Solution, p 2) Write an assembly language program that repeatedly reads numbers from the user until the user types 5.

**Question 5.3–4:** (Solution, p 2) Write an assembly language program that displays 100 copies of the number 0.

**Question 5.3–5:** (Solution, p 2) Write an assembly language program that reads a number  $n$  and displays the powers of two that are less than  $n$ . Your program may assume that  $n$  is positive.

**Question Pseud–1:** (Solution, p 2) Translate the following pseudocode into a close assembly language program equivalent.

Read  $n$ .

**while**  $n = 0$ , **do:**

    Write 1.

    Read  $n$ .

**end while**

Stop.

**Question Pseud–2:** (Solution, p 3) Express in pseudocode an algorithm intended for HYMN to read a number and display its absolute value.

**Question Pseud–3:** (Solution, p 3) Express in pseudocode an algorithm intended for HYMN to read 100 numbers and then to display the maximum among the numbers typed.

Solution 5.3–1: (Question, p 1)	addr	data	translation
	00000	100 11110	READ
	00001	101 11111	WRITE
	00010	110 00101	ADD one
	00011	011 00001	JPOS top
	00100	000 00000	HALT
	00101	000 00001	1

**Solution 5.3–2:** (Question, p 1)

```

READ
STORE n
ADD n
ADD n
ADD n
ADD v3
WRITE
HALT
n: 0
v3: 3

```

**Solution 5.3–3:** (Question, p 1)

```

top: READ
SUB v5
JZER done
JUMP top
done: HALT
v5: 5

```

**Solution 5.3–4:** (Question, p 1)

```

top: LOAD v0
WRITE
LOAD i
SUB one
STORE i
JPOS top
HALT
v0: 0
i: 100

```

**Solution 5.3–5:** (Question, p 1)

```

READ
STORE n
up: LOAD i      # display i
WRITE
ADD i          # double i
STORE i
LOAD n        # repeat if n - i > 0
SUB i
JPOS up
HALT
n: 0
i: 1

```

**Solution Pseud–1:** (Question, p 1)

```
        READ
        STORE n
while:  LOAD n
        JZER done
        LOAD v1
        WRITE
        READ
        STORE n
        JUMP while
        HALT
v1:    1
n:     0
```

**Solution Pseud-2:** (Question, p 1)

Read  $n$ .

**if**  $n > 0$ , **then:**

    Write  $n$ .

**else:**

    Write  $-n$ .

**end if**

Stop.

**Solution Pseud-3:** (Question, p 1)

Initialize  $max$  to  $-128$ .

**repeat** 100 **times:**

    Read  $n$ .

**if**  $n > max$ , **then:**

        Change  $max$  to be  $n$ .

**end if**

**end repeat**

Write  $max$ .

Stop.