

**ACSL CHMOD  
JUNIOR DIVISION**

**PROBLEM:** CHMOD is a command in the UNIX computer system. It is the command used for giving users permissions to access and change files and directories. There are 3 classes of users. They are the owner, the group and others. The permissions given are: read(r), write(w) and execute(x).

The argument of the CHMOD command is a 3-character octal number (ex. 526). When each digit of that number is converted to binary, the binary digits are paired to represent read, write and execute in that order. 526 would convert to 101 010 110.

The first binary conversion gives the user permissions. The second gives the group permissions. The third gives the others permissions. So here, the owner has read and execute permissions and that is represented by r-x. The group has only write permission given by -w-. The others class has read and write permissions as shown by rw-.

Putting all of the above together CHMOD 526 = 101 010 110 = r-x -w- rw-

**INPUT:** There will be 5 lines of input. Each line will contain 3 octal digits.

**OUTPUT:** For each line of input, print the 9-bit binary conversion (in order from left to right with a space between each conversion) and the 9-character permission string (in order from left to right with a space between each conversion). Both outputs must be correct to get the point.

**SAMPLE INPUT**

1. 5, 2, 6
2. 7, 3, 0
3. 4, 1, 5
4. 2, 3, 4
5. 5, 6, 7

**SAMPLE OUTPUT**

1. 101 010 110 and r-x -w- rw-
2. 111 011 000 and rwx -wx ---
3. 100 001 101 and r-- --x r-x
4. 010 011 100 and -w- -wx r--
5. 101 110 111 and r-x rw- rwx

**ACSL****American Computer Science League**

---

**2015 - 2016****Contest #1**ACSL CHMOD  
JUNIOR DIVISION

## TEST DATA

## TEST INPUT

1. 1, 0, 4

2. 7, 7, 2

3. 6, 3, 5

4. 0, 2, 6

5. 5, 2, 7

## TEST OUTPUT

1. 001 000 100 and -- x --- r--

2. 111 111 010 and rwx rwx -w-

3. 110 011 101 and rw- -wx r-x

4. 000 010 110 and --- -w- rw-

5. 101 010 111 and r-x -w- rwx