Project #1
CS 300/500 - Advanced OO Programming in C++

Please complete the following programs. Create a folder on Git called Project1. To submit the assignment, please put the link to your repo with all of your .cpp files in the Project1 folder on Github. **CS300** students, please choose one problem from 1-3 and complete problems 4 and 5. **CS500** students, please choose two problems from 1-3 and complete problems 4 and 5.

1. A common typing error is to place your hands on the keyboard one row to the right of the correct position. Then "Q" is typed as "W" and "J" is typed as "K" and so on. Your task is to decode a message typed in this manner.

   Input
   Input consists of several lines of text. Each line may contain digits, spaces, uppercase letters (except "Q", "A", "Z'), or punctuation shown above [except back-quote (')]. Keys labeled with words [Tab, BackSp, Control, etc.\ are not represented in the input.

   Output
   You are to replace each letter or punctuation symbol by the one immediately to its left on the QWERTY keyboard. Spaces in the input should be echoed in the output.

   Sample Input
   O S, GOMR YPFSU/

   Sample Output
   I AM FINE TODAY

2. A friend of yours has just bought a new computer. Before this, the most powerful machine he ever used was a pocket calculator. He is a little disappointed because he liked the LCD display of his calculator more than the screen on his new computer (haha, as if!). To make him happy, write a program that prints numbers in LCD display style. Be sure to use functions where appropriate.

   Input
   The input contains several lines, one for each number to be displayed. Each line contains integers s and n, where n is the number to be displayed (0 <= n <= 99,999,999), and s is the size in which it shall be displayed (1 <= s <= 10). The input will be terminated by a line containing two zeros, which should not be processed.

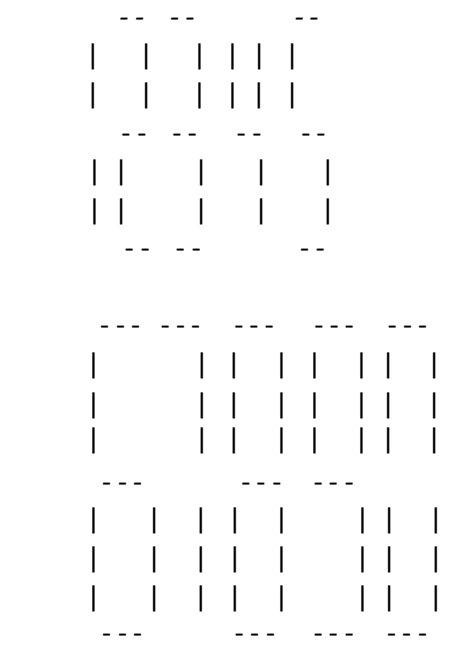
   Output
   Print the numbers specified in the input in an LCD display-style using s "-" signs for the horizontal segments and s "|" signs for the vertical ones. Each digit occupies exactly s + 2 columns and 2s + 3 rows. Be sure to fill all the white space occupied by the digits with blanks, including the last digit. There must be exactly one column of blanks between two digits.

Output a blank line after each number. You will find an example of each digit in the sample output below.

Sample Input

2 12345
3 67890
0 0

Sample Output



3. The reverse and add function starts with a number, reverses its digits, and adds the reverse to the original. If the sum is not a palindrome (meaning it does not give the same number read from left to right and right to left), we repeat this procedure until it does.

   For example, if we start with 195 as the initial number, we get 9,339 as the resulting palindrome after the fourth addition:

195	786	1,473	5,214
+ 591	+ 687	+ 3,741	+ 4,125
786	1,473	5,214	9,339

   This method leads to palindromes in a few steps for almost all of the integers. But there are interesting exceptions. 196 is the first number for which no palindrome has been found. It has never been proven, however, that no such palindrome exists.

   You must write a program (using functions where appropriate) that takes a given number and gives the resulting palindrome (if one exists) and the number of iterations/additions it took to find it.

You may assume that all the numbers used as test data will terminate in an answer with less than 1,000 iterations (additions) and yield a palindrome that is not greater than 4,294,967,295.

Input
The first line will contain an integer N (0 <= N <= 100), giving the number of test cases, while the next N lines each contain a single integer P whose palindrome you are to compute.

Output
For each of the N integers, print a line giving the minimum number of iterations to find the palindrome, a single space, and then the resulting palindrome itself.

Sample Input
3
195
265
750

Sample Output
4 9339
5 45254
3 6666

4. **(CS 300 and CS 500)** Suppose that you have been assigned the task of computerizing the card catalog system for a library. As a first step, your supervisor has asked you to develop a prototype capable of storing the following information for each of 1000 books:
    - The title
    - A list of up to five authors
    - The Library of Congress catalog number
    - A list of up to five subject headings
    - The publisher
    - The year of publication
    - Whether the book is circulating or noncirculating

    Design the data structures that would be necessary to keep all the information required for this prototype library database. Given your definition, it should be possible to write the declaration

    ```
    libraryT libdata; //This is a struct, which often uses the capital T at the end of the name to
            //denote that it is a structure type
    ```

    and have the variable libdata contain all the information you would need to keep track of up to 1000 books. Remember that the actual number of books will usually be less than this upper bound.

Write an additional procedure SearchBySubject that takes as parameters the library database and a subject string. For each book in the library that lists the subject string as one of its subject headings, SearchBySubject should display the title, the name of the first author, and the Library of Congress catalog number of the book.

In your main, populate the data structure with five sample books. Provide the user with a menu to add a new book, search by subject, or display a list of all of the book titles and authors.

5. **(CS 300 and CS 500)** Write a program that plays the game of hangman. In hangman, the computer begins by selecting a secret word at random from a list of possibilities. It then prints out a row of dashes—one for each letter in the secret word—and asks the user to guess a letter. If the user guesses a letter that appears in the word, the word is redisplayed with all instances of that letter shown in the correct positions, along with any letters guessed correctly on previous turns. If the letter does not appear in the word, the player is charged with an incorrect guess. The player keeps guessing letters until either (1) the player has correctly guessed all the letters in the word or (2) the player has made eight incorrect guesses. A sample run of the hangman program is shown below.

   To separate the process of choosing a secret word from the rest of the game, define and implement an interface called randword.h that exports two functions: InitDictionary and ChooseRandomWord. InitDictionary should take the name of a data file containing a list of words, one per line, and read it into an array declared as a static global variable in the implementation. ChooseRandomWord takes no arguments and returns a word chosen at random from the internally maintained array.


Hangman
Welcome to Bangmanl
I will guess a secret word. On each turo, you guess
a letter. If tbe letter is in tbe secret mord, I
will sRow you wbere it appears; if not, a part of
your body gets struDg up on the scaffold. Ibe
object is to guess the word before you are hung.
The mord now looks like this:
Xou have 8 guesses left.
Guess a letter: B
That guess is correct.
The mord now looks like this: ------E-
Xou have 8 guesses left.
Guess a letter: A
There are no A's in the word.
The mord now looks like this: ------E-
Xou have 7 guesses left.
Guess a letter: I
There are no I's in the word.
The mord now looks like this: ------E-
Xou have 6 guesses left.
Guess a letter: O
That guess is correct.
The mord now looks like this: -O----E-
Xou have 6 guesses left.
Guess a letter: S
There are no 6's in the word.
The mord now looks like this: -O----E-
Xou have 5 guesses left.
Guess a letter: T
That guess is correct.
The mord now looks like this: -O---TE-
Xou have 5 guesses left.
Guess a letter: R
That guess is correct.
The mord now looks like this: -O---IER
Xou have 5 guesses left.
Guess a letter: N
There are no n's in the word.
The mord now looks like this: -O---IER
Xou have 4 guesses left.
Guess a letter: P
That guess is correct.
The mord now looks like this: -O-P-HR
Xou have 4 guesses left.
Guess a letter: C
That guess is correct.
The mord now looks like this: Co-P-HR
Xou have 4 guesses left.
Guess a letter: M
That guess is correct.
The mord now looks like this: COMP-TER
Xou have 4 guesses left.
Guess a letter: U
That guess is correct.
Xeu messed the word: OBMPUTRR