COMP500 / ENSE501: Week 7 – Exercise:

EXERCISE NAME: *Spaceman Zuurp*

"Spaceman" is a word-guessing game where the user must guess a secret word, one letter at a time.

Zuurp, an alien, has crashed on planet Earth. They will rebuild their spaceship as the user incorrectly guesses. After each incorrect guess, the next stage of Zuurp's spaceship is built. After incorrectly guessing nine letters, the spaceship is complete! Each correctly guessed letter is revealed in its place in the secret word. When the user guesses the secret word before running out of guesses, the program asks if the user wants to play another round.

The program must be case-insensitive, such that the user can input either a lower-case or upper-case letter.

As the guessing game progresses, Zuurp collects the following data on the user:

- The number of times a secret word is fully revealed.
- The number of times a secret word is not fully revealed.
- The number of correctly guessed letters, for all rounds played.
- The number of incorrectly guessed letters, for all rounds played.

There is one further C programming technique required to fully complete this exercise that you will be unfamiliar with at this point in the course (the end of Week 7): *creating an array of C-Strings*.

Ideally, this program should include five different built-in secret English words, of which one is used at random per round. To do this, you will need an array of C-String's for example:

```
char* secret_words[] =
{
    "ANOTHER",
    "PROGRAMMING",
    "TECHNIQUE",
    "TO",
    "LEARN"
};
```

To utilise this, you may need to do some self-directed study on pointers, **`char`** pointers, storing and accessing C-Strings in an array. These techniques will be covered later in the course.

To avoid this requirement, at the start of the program you can alternatively ask the user for a secret word to use, for example:

```
Game master, what is the secret word?
```

The program should then print enough newlines to clear the screen and hide this input. Allowing input of the secret word will require two players to play the game, one to input the secret word, the other to then play the guessing game. As a programmer, you can take on both roles!

Ensure the program's behaviour is exactly as described below.

At the start of the program, the following output is presented:

```
      -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
         Welcome to SPACEMAN, a word-guessing game!
      -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

    Zuurp, an alien, has crashed on planet Earth! Zuurp
      mistakes you for Earth's leader and demands you
   guess their secret word, one letter at a time. Zuurp's
   people have been monitoring Earth for years, and hence
           know the English language quite well!

     If you don't guess the word quickly enough, Zuurp
   will rebuild their spaceship and leave, but if you do
  manage to guess Zuurp's secret word then they will stay
  to play again! Beware, Zuurp likes numbers, and will be
              recording your data as you guess!

+------------------------------------------------------------+

                            \_/
                            (")
                           ---+---
                            _/ \_

+------------------------------------------------------------+

Zuurp counts 0 incorrect guesses... [ ]

Zuurp's secret word:  _  _  _  _  _  _  _  _

Earth leader?
```

After the introduction is displayed as above, the user can then input a single character response.

Notice that the secret word is displayed using underscore characters. In the example above, the secret word is eight letters in length.

Also notice, Zuurp has counted zero incorrect guesses – this is because the word-guessing game has just started and the user has not made any guesses yet!

Correct guesses reveal the corresponding hidden letters. For example, when the user (Earth's leader) guesses the letter `'m'` the following occurs:

```
+-----------------------------------------------------+

                          \_/
                          (")
                         ---+---
                          _/ \_

+-----------------------------------------------------+

Zuurp counts 0 incorrect guesses... [ ]

Zuurp's secret word:  _  _  _  _  _  M  _  _

Earth leader?
```

The user is then able to make their next guess.

When the user incorrectly guesses the letter `'z'` the following occurs:

```
+-----------------------------------------------------+




   _________________
   \_______________/
                          \_/
                          (")
                         ---+---
                          _/ \_

+-----------------------------------------------------+

Zuurp counts 1 incorrect guess... [ Z ]

Zuurp's secret word:  _  _  _  _  _  M  _  _

Earth leader?
```

Notice the first stage of Zuurp's spaceship has been built due to the incorrect guess of the letter `'z'`!

This pattern of interaction continues where the user is asked for their guess, and Zuurp either uncovers the correct, corresponding letter from their secret word, or they add the incorrectly guessed letter to their list of incorrect guesses.

Any time the user's guess is not a letter, the program must respond as follows:

```
Earth leader? 3

Zuurp says, "Qzwopp! 3 is not a letter!"

Earth leader?
```

Another example is as follows:

```
Earth leader? #

Zuurp says, "Qzwopp! # is not a letter!"

Earth leader?
```

Non-letter guesses do not count towards the user's tally of incorrectly (or correctly) guessed letters!

Note, "Qzwopp!" means "invalid input" in Zuurp's language!

After eight incorrect guesses, the user only has one incorrect guess left:

```
+-----------------------------------------------------------+

        .........
      ..         ..
 ____/_____________\____
/__O__O__O__O__O__O__O__\
   \_|_|_|_|_|_|_|_|_|_/
                        \_/
                        (")
                       ---+---
                        _/ \_

+-----------------------------------------------------------+

Zuurp counts 8 incorrect guesses... [ Z X V B D F G H ]

Zuurp's secret word:  S  _  A  C  _  M  A  N

Earth leader?
```

In the example above, the user has correctly guessed the letters **'A'**, **'C'**, **'M'**, **'N'**, and **'S'**. They have incorrectly guessed eight letters: **'Z'**, **'X'**, **'V'**, **'B'**, **'D'**, **'F'**, **'G'** and **'H'**. At this point, the user is only allowed one final incorrect guess!

If the user is unable to guess correctly after their ninth incorrect guess, the following is displayed:

```
+-------------------------------------------------------+

        .........
      ..   \_/   ..
 ____/_____(")_____\____
/__O__O__O__O__O__O__O__\
   \_|_|_|_|_|_|_|_|_/

+-------------------------------------------------------+

Zuurp counts 9 incorrect guesses... [ Z X V B D F G H J ]

Zuurp says, "My secret word was..."

 S  P  A  C  E  M  A  N

      -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

                Zuurp is ready to go home!

      -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

Zuurp says, "Do you want to try again?"

Earth leader?
```

The user must respond with either `'y'` or `'n'`, in either upper- or lower-case.

When the user's response is not the letter `'y'` or `'n'`, the program must respond as follows:

```
Zuurp says, "Do you want to try again?"

Earth leader? q

                          \_/
                          (")
                         ---+---
                          _/ \_

Zuurp demands, "You must choose 'Y' or 'y' to play!"

Zuurp says, "Do you want to try again?"

Earth leader?
```

When the user response with yes, the following occurs and a new round starts:

```
      -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

       Something went wrong with Zuurp's spaceship...

     Zuurp returns to planet Earth to rebuild again...

      -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

+---------------------------------------------------------+

                          \_/
                          (")
                         ---+---
                          _/ \_

+---------------------------------------------------------+

Zuurp counts 0 incorrect guesses...

Zuurp's secret word:  _  _  _  _  _  _  _  _

Earth leader?
```

When the user responds with no, the following message is displayed, and the program then ends:

```
Zuurp says, "Do you want to try again?"

Earth leader? n

      -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

        Zuurp left planet Earth for good this time!

      -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

  0 words fully revealed.
  1 word not fully revealed.
  5 letters correctly guessed.
  9 letters incorrectly guessed.

Beware! Zuurp will be reporting this data to their leader!

      -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

Press ENTER to quit...
```

To finally exit the program, the user must press the “ENTER” key. Note, the final statistics are based upon data collected as the user interacts with the program. This output must be plural-sensitive.

When the user fully reveals the secret word, the following occurs:

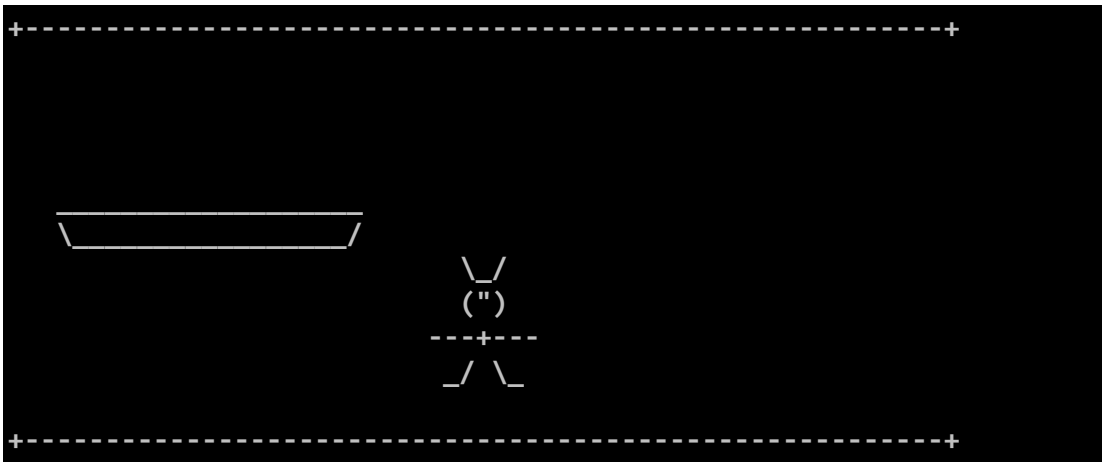
```
Zuurp says, "My secret word was..."

 S  P  A  C  E  M  A  N

      -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

    Xzoopz!! That means 'well done' on Zuurp's world!

      -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

                          \_/
                          (")
                         ---+---
                          _/ \_

Zuurp says, "Do you want to try again?"

Earth leader?
```

After successfully guessing the secret word, Zuurp asks if the user wants to try again.

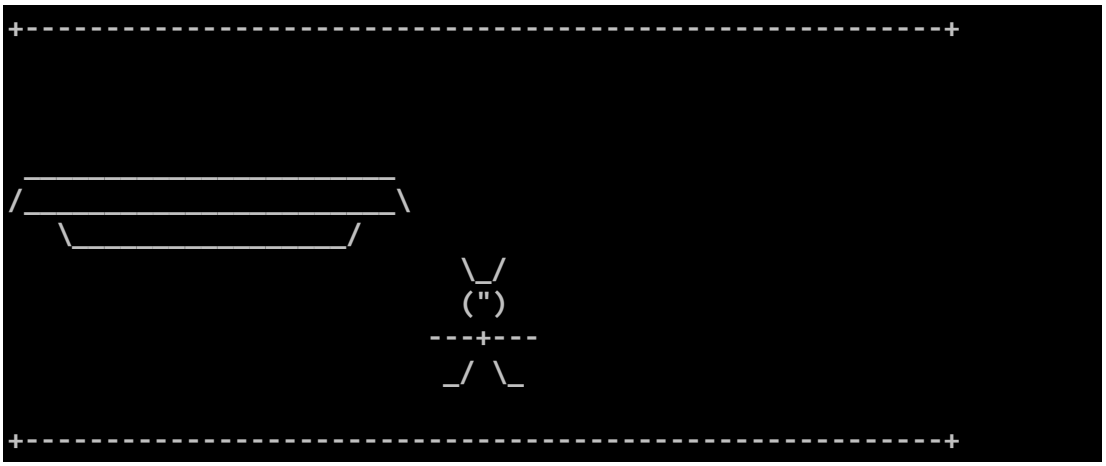
Starting a new round will cause Zuurp to start building his spaceship again from scratch!

Remember, incorrect letter guessing causes each stage of Zuurp's spaceship to be built. The nine stages are shown on the following pages.

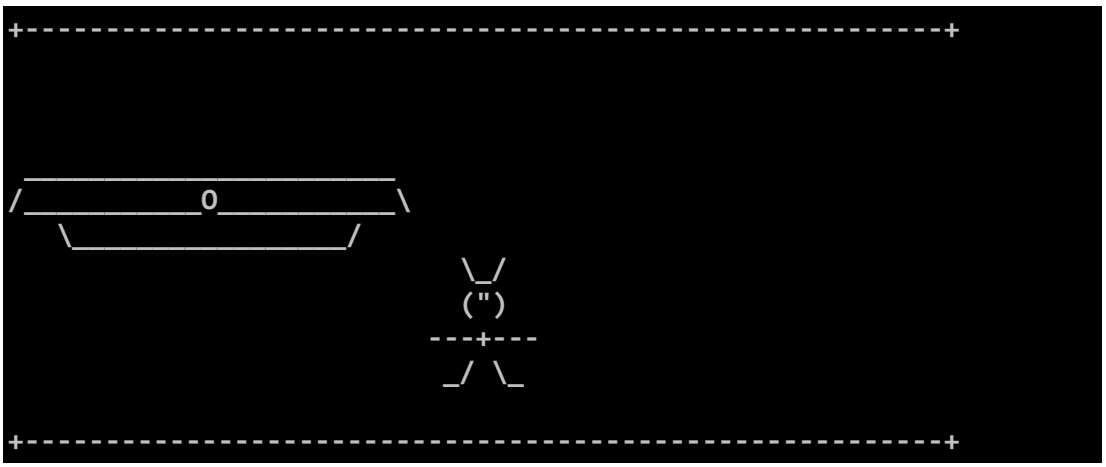
The spaceship is built in stages as follows. After one incorrect guess, Zuurp builds the underside:



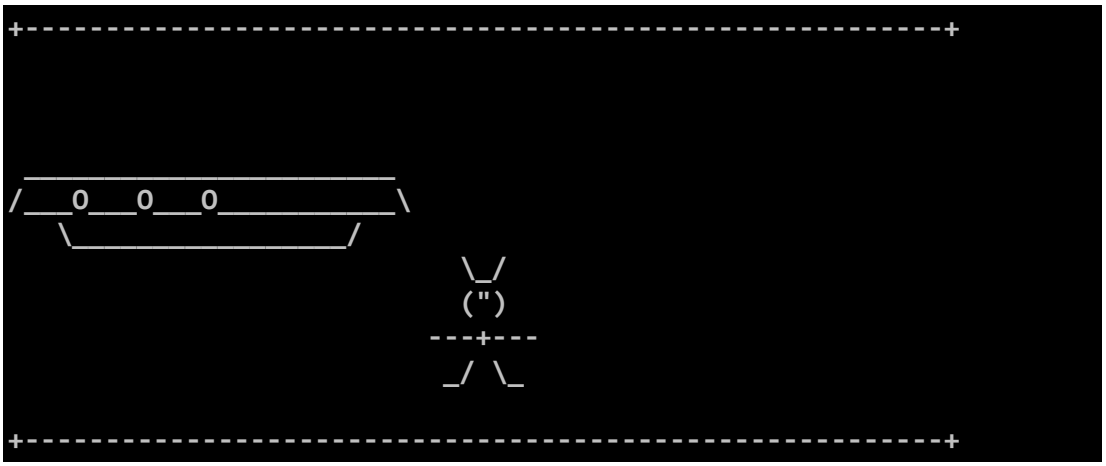
After two incorrect guesses, Zuurp builds the middle ring:



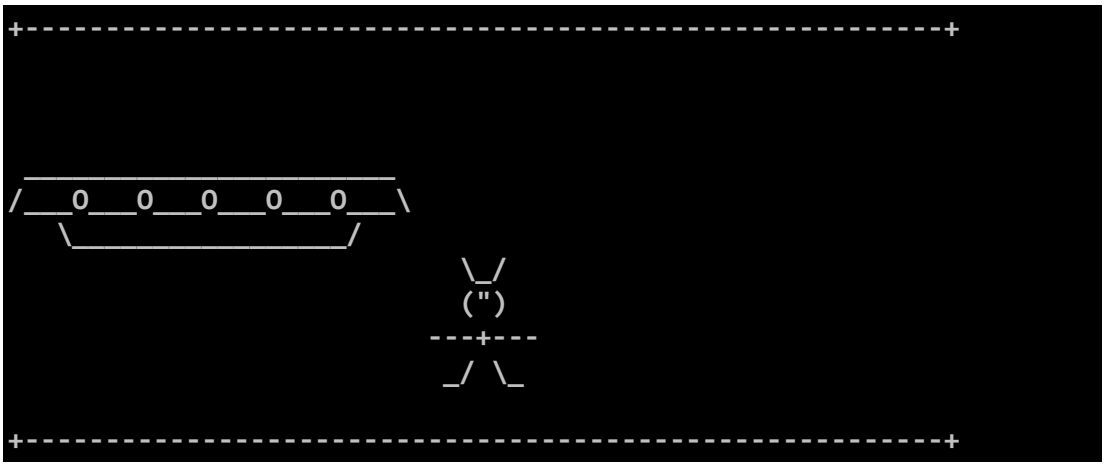
After three incorrect guesses, Zuurp builds the middle sensor:



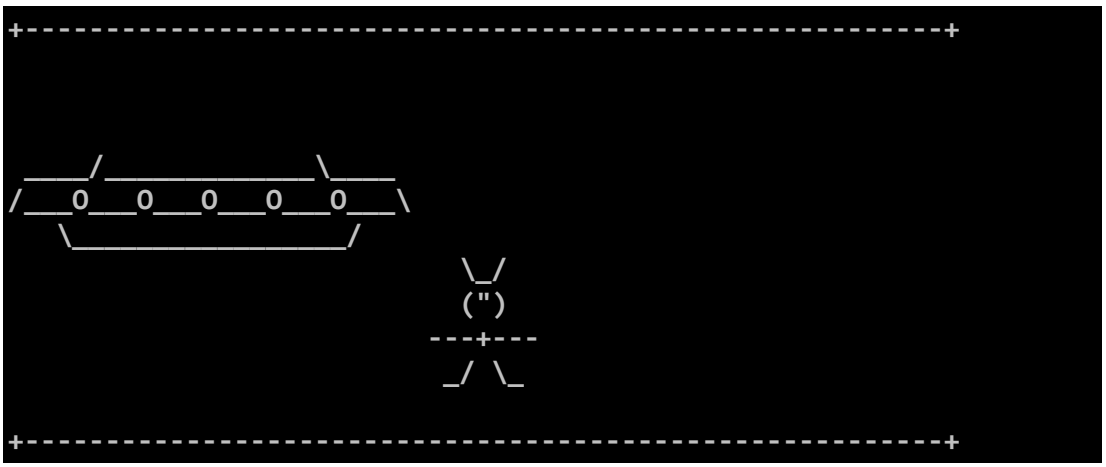
After four incorrect guesses, Zuurp builds the two left sensors:



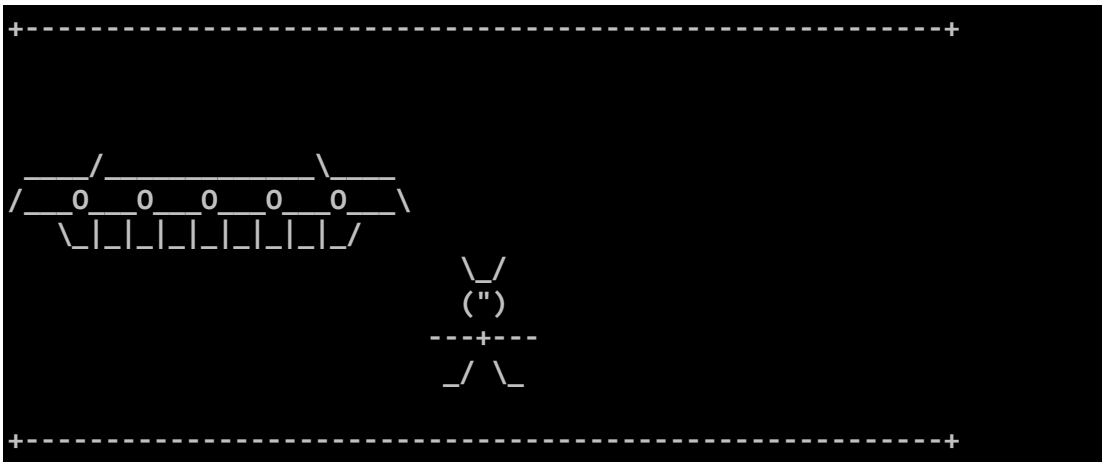
After five incorrect guesses, Zuurp builds the two right sensors:



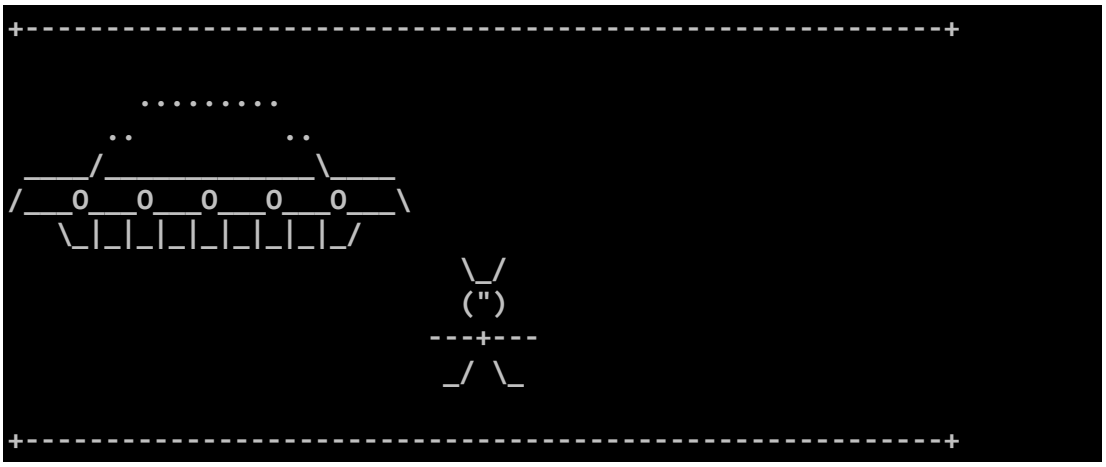
After six incorrect guesses, Zuurp builds the dome connection points:



After seven incorrect guesses, Zuurp builds the underside panels:



After eight incorrect guesses, Zuurp builds the dome:



After nine incorrect guesses, Zuurp gets into their spaceship:

At this point, the round is over, and the user has failed to guess the secret word!

Based upon the program requirements set out above, design, implement, test and debug the "Spaceman" program using C.

Ensure your implementation demonstrates the following programming techniques and good practices:

- Follows good programming standards for:
  - Naming.
  - Commenting.
  - Code layout white space.
- Sequence.
- Selection.
- Repetition.
- Variables.
- Arrays.
- C-Strings.
- Declare and define at least **seven** modular functions in addition to the **`main`** function.
  - At least one of your functions must have a parameter.
  - At least one of your functions must return a value.
- Ensure your C source code successfully compiles.
- Do not:
  - use global variables.
  - use **`goto`**.
  - Corrupt the program's memory.

Robustly test your program with a variety of inputs and ensure the resulting program outputs are as described in the requirements above.