Strings Workshop!

Workshop 10 (worth 3% of your final grade)
URL: https://github.com/Seneca-144100/IPC-WS10

In this workshop, you will work with strings. Part A will introduce us to working with strings and the string library. In Part B you will create a simple game using strings.

LEARNING OUTCOMES

Upon successful completion of this workshop, you will have demonstrated the abilities

- Allocate storage for one or more strings
- Use standard input facilities to store text entered by users
- Use standard output facilities to display text data to the user
- Modify string memory directly using the [ ] operator
- Use string library functions to manipulate strings

SUBMISSION POLICY

Your workshops are divided in two sections; in_lab and at_home.

The “in_lab” section is to be completed **during your assigned lab section**. It is to be completed and submitted by the end of the workshop. If you do not attend the workshop, you can submit the “in_lab” section along with your “at_home” section (a 20% late deduction will be assessed). The “at_home” portion of the lab is **due the day before your next scheduled workshop**

All your work (all the files you create or modify) must contain your name, Seneca email and student number.

You are responsible for regularly backing up your work.

IN-LAB: READ AND SEARCH A TEXT FILE (70%)

Download or clone workshop 10 from https://github.com/Seneca-144100/IPC-WS10

There are 9 tasks in the code labelled “TODO” as comments. When you complete the tasks, your program should give the output below:

Output Sample 1:

```
default name is: default
Enter a name 100 characters or less: Millhouse
could not copy name, was too long!!
input is: Millhouse
name is: default
```

Output Sample 2:

```
default name is: default
Enter a name 100 characters or less: Millhous
name was successfully changed!
input is: Millhous
name is: Millhous
```

For submission instructions, see the SUBMISSION section below.

IN_LAB SUBMISSION:

To test and demonstrate execution of your program use the same data as the output example above or any information needed….

If not on matrix already, upload your `w10_in_lab.c` to your matrix account. Compile and run your code and make sure everything works properly.

Then run the following script from your account:

```
~profname.proflastname/submit ipc_w10_in_lab <ENTER>
```

and follow the instructions.

AT_HOME: WRITE IN A TEXT FILE (20%)

After completing the in_lab section, go ahead and start a new file, w10_at_home.c and create a game with the following specifications.

Create a game that lets the user enter their name and then guess a password until they run out of lives. The users's remaining lives are the characters in their name.

Get the user to enter their name up to 8 characters long and store it. Each time they incorrectly guess the password one character will be subtracted from their name.

Make a backup of the user's name in a string called backup (you'll need it later). Use strcpy to make the copy.

Determine the length of the user's name using the strlen function. While the user's name has more than 0 characters remaining, allow them to guess the password.
The secret is that the password is generated by taking the user's name and subtracting 1 from the ASCII code for each character. For example, if the user enters the name "Bob" the password will be "Ana". After the user enters their name, make a copy of the user's original name using strcpy and store it in another string. Write a loop to subtract 1 from each character in the password, stopping when the length is reached.

Make sure you allocate a string to store the users's guess. Then use strcmp to determine if each guess is correct. Each time they fail to guess the password, their name is shortened by 1 character by writing a terminating null (0) in the position one less than the current length. Create a function called trim that accepts the string as an argument and shortens the string by 1 character. Notice you do not need to pass the string's length as you can use strlen within the trim function to determine the length of the string.

Compare the users's guess with the generated password by using strcmp.

If the user wins by guessing the password. display a message including their original unmodified name (remember, you should have backed it up).

If the user's name reaches length 0, they lose the game.

Output Sample 1:

```
Password Game
=============
Enter your name (up to 8 characters): THRILLHO
WELCOME THRILLHO
Guess the password (up to 8 characters) THRILLHO: elf
FALSE: Guess the password (up to 8 characters) THRILLH: hobbit
```

```
FALSE: Guess the password (up to 8 characters) THRILL: beets
FALSE: Guess the password (up to 8 characters) THRIL: bears
FALSE: Guess the password (up to 8 characters) THRI: battle
FALSE: Guess the password (up to 8 characters) THR: star
FALSE: Guess the password (up to 8 characters) TH: galactic
FALSE: Guess the password (up to 8 characters) TH: dog
FALSE: Guess the password (up to 8 characters) T: cat
Thou has failed.
Game Over!
```

Output Sample 2 (this time, somebody has told our player the secret!):

```
Password Game
=============
Enter your name (up to 8 characters): THRILLHO
WELCOME THRILLHO
Guess the password (up to 8 characters) THRILLHO: password
FALSE: Guess the password (up to 8 characters) THRILLH: oops
FALSE: Guess the password (up to 8 characters) THRILL: SGQHKKGN
You got it, THRILLHO!!!
Game Over!
```

AT-HOME REFLECTION (10%)

Please provide brief answers to the following questions in a text file named `reflect.txt.`

1) Why do you need to be careful with the terminating null character in strings?
2) What can happen if the terminating null is not inserted or if it is overwritten accidentally?

AT_HOME SUBMISSION:

To test and demonstrate execution of your program use the same data as the output example(s) above.

If not on matrix already, upload your `reflect.txt` and `w10_at_home.c` to your matrix account. Compile and run your code and make sure everything works properly.

Then run the following script from your account:

```
    ~profname.proflastname/submit ipc_w10_at_home <ENTER>
```

and follow the instructions.