CSC1300: LAB 10

PASSWORD VALIDATION

CONCEPTS

- Pointers
- Dynamnic Memory Allocation
- Character-testing functions

DIRECTIONS

Create a program named **lab10.cpp** that will have the main function and a testPassWord() function. It may be easiest to view the Sample Output near the bottom and then read the directions.

Main Function

- Ask the user "`How many characters should the password be (must be at least 4)?`"
- Read in the number into an integer variable.
- Validate the input using the methods taught in lab 9. Remember to validate data type as well as making sure the number is in the valid range.
- Now add one to this number. This will be the size of the character array (c-string) that you will now **dynamically allocate**. You added one so that there is space for the null terminator. **If you do not dynamically allocate this array, 20 points will be deducted for this assignment!**
- Then, ask the user for a password but give the user these exact directions:
  "`Enter a password consisting of exactly X characters.`
  `You must have at least one capital letter, one lowercase letter, one digit,`
  `and one punctuation character (examples: !#$%.?@).`
  `The password can't contain any spaces.`"
  X is the number that they entered in for the number of characters their password should be.
- Read in their password into the dynamically allocated c-string array and then validate the user input to make sure it is exactly the correct number of characters. If not, force them to enter it again. (this should be a loop)
- Then, tell the user "Please wait – your password is being verified".

- Then, make the output delay moving forward for 3000 milli-seconds. To do this, you will need to #include two libraries in this source file. One is <windows.h> and the other is <unistd.h> and this is so the sleep() function that you will use will work on both windows and mac/linux systems. Then, the actual programming statement to make the delay happen is: `Sleep(3000);`
- Then, call the testPassWord function, sending the dynamically allocated c-string to the function. This function should return a true if the password is valid and false if it is not valid. If the function returns false, print "----------Invalid password!----------". Then, the program should allow the user to enter in a different password until they are able to come up with a password that is valid.
- When the password is valid, print "Yay! You came up with a valid password!"

testPassWord function

The testPassWord() function will have one parameter, which is a pointer to a char. The function should return a bool.

The function will test the password to see if there is at least one uppercase letter, one lowercase letter, one digit, one punctuation character, and that the password doesn't contain any spaces.

Every time you access an array element, make sure it is in pointer notation instead of array notation! 10 points will be removed from your submission if you do not use pointer notation in this function.

If the password doesn't contain one (or more) of these things, then an appropriate error message should be printed. For example, if there were no uppercase letters in the password, the function should print "`Oops! You didn't enter at least one uppercase letter!`".

If the password is valid, this function should return true and false otherwise.

SAMPLE OUTPUT

User input is highlighted in **yellow**.

```
How many characters should the password be (must at least 4)?  abcd

There was a problem with what you entered.

How many characters should the password be (must at least 4)?  3

There was a problem with what you entered.

How many characters should the password be (must at least 4)?  5

Enter a password consisting of exactly 5 characters.
You must have at least one capital letter, one lowercase letter, one digit,
and one punctuation character (examples:  !#$%.?@).
The password can't contain any spaces.
aeiou
Please wait - your password is being verified

Oops!  you didn't enter at least one uppercase letter!
Oops!  You didn't enter at least one digit!
Oops!  You didn't enter at least one punctuation mark!
----------Invalid password!----------

Enter a password consisting of exactly 5 characters.
You must have at least one capital letter, one lowercase letter, one digit,
and one punctuation character (examples:  !#$%.?@).
The password can't contain any spaces.
AEIOU
Please wait - your password is being verified
```

```
Oops!  You didn't enter at least one lowercase letter!
Oops!  You didn't enter at least one digit!
Oops!  You didn't enter at least one punctuation mark!
----------Invalid password!----------

Enter a password consisting of exactly 5 characters.
You must have at least one capital letter, one lowercase letter, one digit,
and one punctuation character (examples:  !#$%.?@).
The password can't contain any spaces.
93949
Please wait - your password is being verified

Oops!  you didn't enter at least one uppercase letter!
Oops!  You didn't enter at least one lowercase letter!
Oops!  You didn't enter at least one punctuation mark!
----------Invalid password!----------

Enter a password consisting of exactly 5 characters.
You must have at least one capital letter, one lowercase letter, one digit,
and one punctuation character (examples:  !#$%.?@).
The password can't contain any spaces.
#$%@!!!!

There was a problem with what you entered.
Maybe you didn't enter 5 characters?  Please try again:  #@$#!
Please wait - your password is being verified

Oops!  you didn't enter at least one uppercase letter!
Oops!  You didn't enter at least one lowercase letter!
Oops!  You didn't enter at least one digit!
----------Invalid password!----------

Enter a password consisting of exactly 5 characters.
You must have at least one capital letter, one lowercase letter, one digit,
and one punctuation character (examples:  !#$%.?@).
The password can't contain any spaces.
a4! W
Please wait - your password is being verified

Oops!  Your password contains a space!
----------Invalid password!----------

Enter a password consisting of exactly 5 characters.
You must have at least one capital letter, one lowercase letter, one digit,
and one punctuation character (examples:  !#$%.?@).
The password can't contain any spaces.
a4!Wh
Please wait - your password is being verified

Yay!  You came up with a valid password!
```

WHAT TO TURN IN

ZIP & upload the following files to the ilearn submission folder:

- lab10.cpp