DT211C/1 Programming Assignment #2

Due Date: Sunday, March 10th, 2019 (11.59pm)

You are required to develop a program that will perform security authorisation based on access codes. The access codes are 4 single-digit integer numbers between 0-9. The program should allow a user to enter a code, encrypt the number and compare it to an authorised access code. The program should also allow the user to decrypt an encrypted code.

When your program begins executing, the default authorised access code is **4523** (encrypted form of 1234 – see encryption algorithm below). This code must be stored in a 1-D array called *access_code* and should not be changed. You should use a different 1-D array to read the code entered by the user.

Your program should be menu-driven and must display a simple menu when run. The menu should include the following options:

1. Enter code
2. Encrypt code and verify if correct (i.e., matches authorised access code)
3. Decrypt code
4. Display the number of times the code was entered (i) Successfully (ii) Incorrectly
5. Exit Program

Note:

- Each menu option must be implemented in a separate function, i.e. modularized.
- All functions must pass parameters using **Pass by Reference**. Do NOT pass parameters using pass by value.
- All reading and writing to/from arrays must use pointer notation - not subscript notation (i.e. using [ ])

Requirements:

(Each implemented in a separate function):

1. Enter 4 single-digit integers. Perform any necessary validation (error-checking).
2. Encrypt the code entered. You should use the following algorithm to encrypt the 4 single-digit integers:

Encryption Algorithm:

- Swap the 1st number with the 3rd number.
- Swap the 2nd number with the 4th number.
- Add 1 to each number.
- If any number has a value equal to 10, change this value to 0.

Compare the encrypted code with the access code (4523) stored in the 1-D array called *access_code*. If the two codes match, display a message saying “Correct Code entered”. If the two codes do not match, display a message saying “Wrong Code entered” .

3. Decrypt an encrypted code. You should use the following algorithm to decrypt the encrypted code only:

   Decryption Algorithm:
   - Subtract 1 from each number.
   - If any number has a value equal to -1, change this value to 9.
   - Swap the 1st number with the 3rd number.
   - Swap the 2nd number with the 4th number.

4. Count and display the number of times a user entered a code successfully & incorrectly in each run of the program.
5. The program should terminate gracefully.

Features to include:

- After each option has finished, your program should return to the main menu and allow the user to select another option.
- The user should only be allowed to encrypt their code (i.e. select option 2) if the code is NOT already encrypted.
- The user should only be allowed to decrypt their code (i.e. select option 3) if the code IS already encrypted.
- Only encrypted codes should be compared with the access-code (option 2).
- Display appropriate error messages to handle any errors.

Submission details:

1. Submit your program using Webcourses. This must be submitted on or before Sunday, March 10th, 2019 (11.59pm).
2. Extra marks will be awarded for well written code (comments, layout, indentation, whitespace, good use of brackets, etc.,).

Note: You are required to demo your program in the lab within 2 weeks following submission. Failure to demo your program will result in a zero-mark being awarded.

Late submissions (within 1 week) will be marked out of 50%. No submissions accepted after 1 week and a zero-mark awarded.

NB - This is an individual assignment and **NOT** a group one. Do your own work and do not copy the work from a fellow student. Any assignments submitted which are found to be copied will result in both students failing the assignment. Also, any students found to provide their work to other students will fail this assignment. Suspected copying or plagiarising of another piece of work will be dealt with by the DIT code of ethical conduct.