SWE20004 -Technical Software Development

Assignment 1

(This assignment is worth 5% of the subject's total assessment marks)

Due Date: Friday 6th September 2019 at 11:59 pm

Submission instructions

Submit a soft copy of required documents through Canvas.

Introduction:

This assignment requires the knowledge of variables, constants, stream input / output, output formatting, assignment statements, expressions as well as the sequence, selection and loops

You should complete **all three stages** of the problem. In addition to the answer to the assignment question, you need to submit **Task 2.11** from week 2, **Task 3.8** from week 3 and **Task 4.9** from week 4 with this submission.

Problem

Assume you are working as a programmer in a communication field. Your task is to alter the original data before transmission. All of their data is an **integer** that contains the number of digits between two and eight (inclusive), thus the possible integers are between 10 and 99999999. You have to read in an integer and complete the following stages.

Stage 1:

Your task is to modify the data for transmission according to the following set of rules.

1. If the number is a two digit number you swap the positions of the numbers (eg: if the number is 56, altered number is 65).
2. If the number is a three digit number, digits at positions 1 and 3 are swapped. (eg: number is 123, altered number is 321)
3. If the number is four digits or above, the following rules apply.
   Replace each digit by the remainder after that digit is multiplied by 3 is divided by 10.

Stage 2: Extra protection (done only on 8 digits numbers – if the user chooses this option)

Eight digit data contains more valuable information, so you are asked to enable extra protection if the user chooses to.

The extra protection is done on **modified number in stage one** by swapping the digit in position 1 with digit in position 8, digit in position 2 with digit in position 7, digit in position 3 with digit in position 6, digit in position 4 with digit in position 5.

If the modified number in stage 1 is as shown below:

2	3	4	5	6	7	8	9

The number after stage 2 will look like:

9	8	7	6	5	4	3	2

Stage 3: (Decoding)

Once the encoding stages are over, write the codes to recover your original data **from the encoded data**.

Other requirements:

- This assignment must be written in C++ in a procedural language format.
- Your code must have appropriate header(multiline/block) comments including your name and student number, the name of the .cpp file, the purpose of the program, brief explanations of variables and explanations of any code, which is not obvious to another programmer, summarising the input, output and local variables as well as expressions used in your program and test data.
- Include inline (single-line) comments throughout the program describing important statements.
- Use appropriate and descriptive variable following the naming rules and conventions.
- Write a brief (no more than several pages) report, which illustrates your program design (algorithm or flowchart, identification of variables, constants) and include evidence of testing – screen shots or pasted output text of several tests, and the contents of the .cpp file
- Marks will be allocated depending on the amount of original work submitted. Marks will be deducted or zero marks will be awarded for plagiarised and/or un-attributed work.

Assignment submission:

Submissions through **Canvas** must be made on or before the due date/time.

Each submission should have two files.
1. A report (name of the report should be with your student number, eg: 1012546_assignment1.docx)

This report will be used for plagiarism check using turnitin software. **20% of marks will be deducted if this report is missing for plagiarism check.** Report must (***.doc/docx, .pdf*** or ***.rtf*** format – use SWE20004_AssignmentReportTemplate) contain:

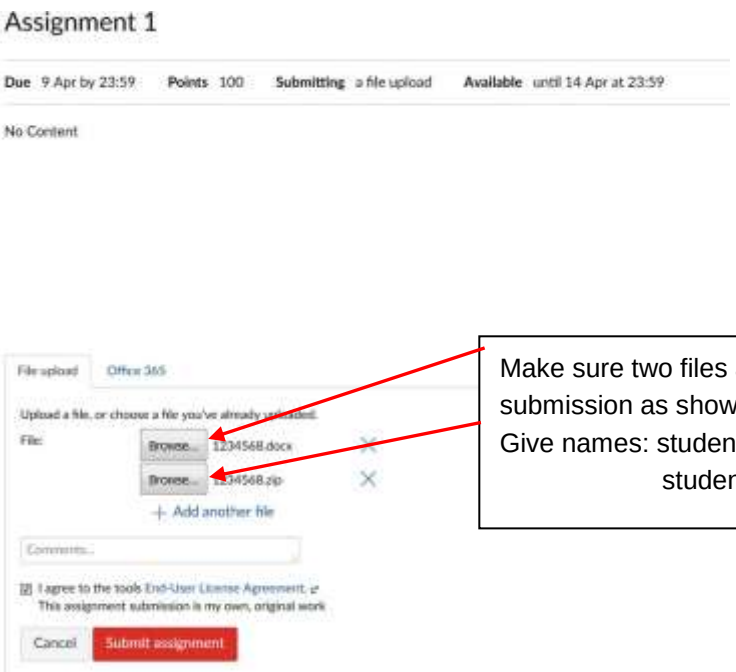
- Description of the problem,
- Description of inputs and outputs,
- A description of the algorithm(s) used in pseudocode or a flowchart
- A copy of the contents of the **.cpp** file
- Pasted **text output** or **screen shots** of the working program resulting from the testing of the program

2. A ***.zip*** file (name of the zip file should be your student number, eg: 1012546_assignment1.zip) containing:

a) The actual program (.**cpp** source code) with comments.
   - The name of your **.cpp** file must be **your student number.cpp (eg**. 386123.cpp) for the assignment question.

b) Task2.11.cpp
c) Task3.8.cpp
d) Task4.9.cpp
   - If you are using Visual Studio for this assignment, do not include the solution files, folders or **exe** files.

Submissions larger than 5 MB will not be accepted.



Make sure two files are selected separately for submission as shown in this figure.
Give names: studentid_Assignment1.docx
studentid_Assignment1.zip

Rubrics

Requirement	Weight (%)	Mark awarded
Program specification and design: (1) Specification of what the purpose and functıonality of the program is (2) Design presented as pseudocode or flowchart	10	
Documentation: (1) Header comments describing (a) the purpose and function of the program (b) Subject and assignment details (c) personal details of the author. (2) Inline comments where appropriate to describe crucial program statements	5	
Coding: Program includes the following elements in order to meet the requirements - Necessary preprocessing directives - Namespace specification - Correct main() function header - Opening and closing braces for the body of main() - Return statement to terminate main() and program - Variable and constant declarations - Input and Output statements - Appropriate use of formatting - Processing including assignment statements, expressions, formulae and calculations as necessary - Appropriate use of sequence, selection - Correct syntax - Correct logic - No runtime errors - Appropriate use of identifier naming rules and conventions - Use of appropriate indentation - sufficient number od appropriate test cases performed and corresponding screen shots provided as evidence	55	
Lab tasks: Task 2.11 from week 2, Task 3.8 from week 3 and Task 4.9 from week 4.	30	
Deductions: Marks will be deducted accordingly for invalid submission of required documents such as missing files, corrupt files, incorrect file formats, use of programming language(s) other than C++ and late or non-submission		
Total Assignment mark (out of 100)	100	
Contribution to unit mark (out of 5)	5	

Screenshots showing working program:

```
Enter a number
1
Not a valid input, re enter the number
-123
Not a valid input, re enter the number
12
Number contains 2 digits

The encoded number is 21

The decoded number is 12
```

```
Enter a number
123
Number contains 3 digits

The encoded number is 321

The decoded number is 123
```

```
Enter a number
2345
Number contains 4 digits

Encrypted number is 6925
Original number is 2345
```

```
Enter a number
9876543
Number contains 7 digits

Encrypted number is 7418529
Original number is 9876543
```

```
Enter a number
12345678
Number contains 8 digits

Do you want to do second level of encryption, enter yes or no? no
Encrypted number is 36925814
Original number is 12345678
```

```
Enter a number
12345678
Number contains 8 digits

Do you want to do second level of encryption, enter yes or no? yes
Encrypted number is 41852963
Original number is 12345678
```

Need help?

Talk to your tutor or visit programming Help Desk in ATC620 (8.30 am – 6.30 pm Monday to Friday)

TSD Tutors - Help Desk Times:

Syeda: Mon 1.30 -2.30, Wed 1.30-2.30, Thu 1.30-2.30 pm and Friday 9.30 - 10.30 am
Srikanth: Monday 4.30 -6.30 pm
Anika: Thursday12:30 - 2:30 pm
Kai: Monday: 11:30 am - 1:30 pm
Michael: Friday 2:30 - 4:30pm
Gavin: Wednesday 8.30 – 10.30 am
Rida: Tuesday 12 - 2 pm