SWE20004 -Technical Software Development

Assignment 2

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

Due Date: Wednesday 9th October 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, sequence, selection (if/switch statements), loops, arrays, functions etc.

You should complete the **problem**.
Need to submit **Task 5.3 and Task 5.4** from week 5, **Task 7.3 and Task 7.4** from week 7.

Problem

Some credit card companies use Luhn algorithm to detect invalid credit card numbers. The key thing in this algorithm is to find a check digit, which should be the rightmost digit in any credit card number.

In this task, you have to implement a version of Luhn algorithm using the instructions given below.

Step1: Read in credit card number as a series of digits into an array

Ask the user to enter series of one digit positive numbers into a one dimensional array, the number -1 is used to indicate the end of series, no need to read in -1 into the array. The maximum size of the array is set to 20.

Step2: Find sum1 (main should call a function passing array and size)

Ignoring the last `check digit` (right most digit) of the credit card number, and moving left, double the value of every second digit and find the sum of these doubled numbers. If the result of the doubling operation is a two digit number, you should add the digits of the doubled number before finding the sum. You must printout the numbers for sum1, these should be in the order (from left to right) as it appears on credit card.

Step 3: find sum2 (main should call a function passing array and size)

Find the sum of all other numbers (last digit is not included in this sum as well). You must printout the numbers for sum1, these should be in the order (from left to right) as it appears on credit card.

Step 4: Calculate check sum

Compute the total of sum1 and sum2 and multiply the result by 9, `checksum` is found by extracting the rightmost digit.

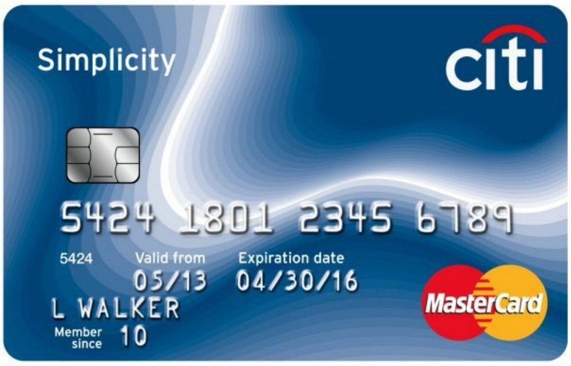


Image courtesy: http://echeck.org/citi-simplicity-credit-card-review/

5	4	2	4	1	8	0	1	2	3	4	5	6	7	8	9
5+5=10 1+0=1		2+2=4		1+1=2		0		2+2=4		4+4=8		6+6=12 1+2=3		8+8=16 1+6=7	sum1=1+4+2+0+4+8+3+7 = 29
	4		4		8		1		3		5		7		sum2 = 4+4+8+1+3+5+7 = 32
															Total=29+32 = 61

61 × 9 = 549

9 is the check sum digit.

Other requirements:

- This assignment must be written in C++.
- **You must use integer array for this problem, vector is not allowed.**
- 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 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_assignment2.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:

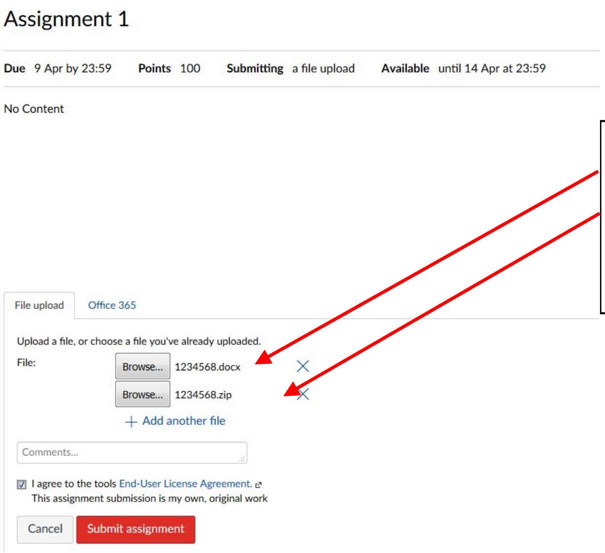
- Description of the problem,
- Description of inputs and outputs,
- A description of the algorithm(s) used in pseudocode or a flowchart
- A copy of the contents of the **.cpp** file (copy and paste the code not the screenshot of the code)
- Pasted **text output** or **screen shots** of the working program 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) Task5.3.cpp
c) Task 5.4.cpp
d) Task7.3.cpp
e) Task7.4.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.



Assignment 1

Due 9 Apr by 23:59 Points 100 Submitting a file upload Available until 14 Apr at 23:59

No Content

File upload Office 365

Upload a file, or choose a file you've already uploaded.

File: Browse... 1234568.docx
Browse... 1234568.zip
+ Add another file

Comments...

☑ I agree to the tools End-User License Agreement.
This assignment submission is my own, original work

Cancel Submit assignment

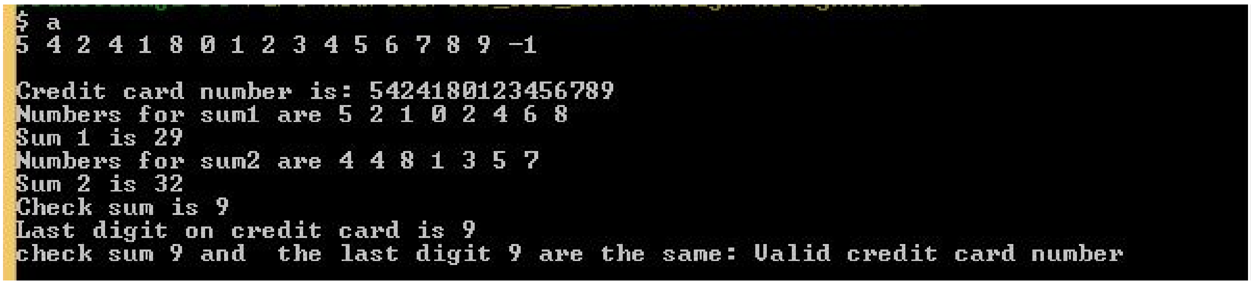
Make sure two files are selected separately for submission as shown in this figure.
Give names: studentid_Assignment2.docx
studentid_Assignment2.zip

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 - Proper use of arrays and functions - 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 - overall coding structure and efficiency - sufficient number of appropriate test cases performed and corresponding screen shots provided as evidence	50	
Lab tasks: Task 5.3.cpp and Task 5.4 (code and screenshots of output window)	10	
Task 7.3 (code and screenshots of output window)	5	
Task 7.4 (code and screenshots of output window)	20	
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 6)	5	

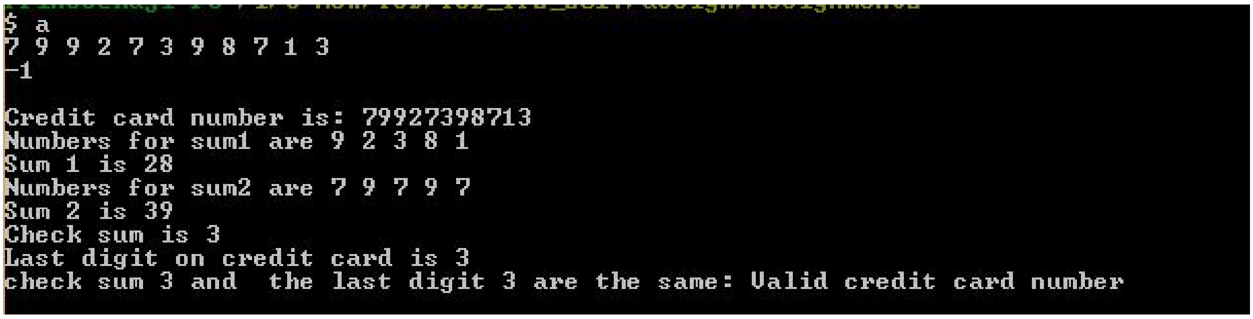
Screenshots showing working program:

Screen shot for the card number shown above



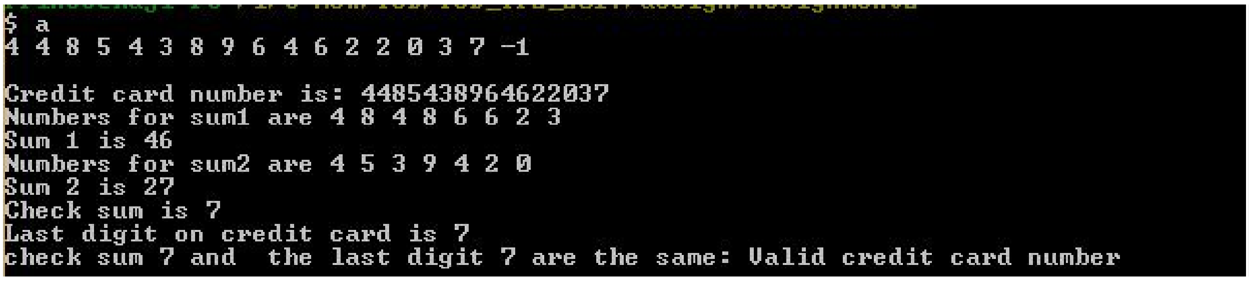
```
$ a
5 4 2 4 1 8 0 1 2 3 4 5 6 7 8 9 -1

Credit card number is: 5424180123456789
Numbers for sum1 are 5 2 1 0 2 4 6 8
Sum 1 is 29
Numbers for sum2 are 4 4 8 1 3 5 7
Sum 2 is 32
Check sum is 9
Last digit on credit card is 9
check sum 9 and  the last digit 9 are the same: Valid credit card number
```

Screen shot for the card number shown on wiki page

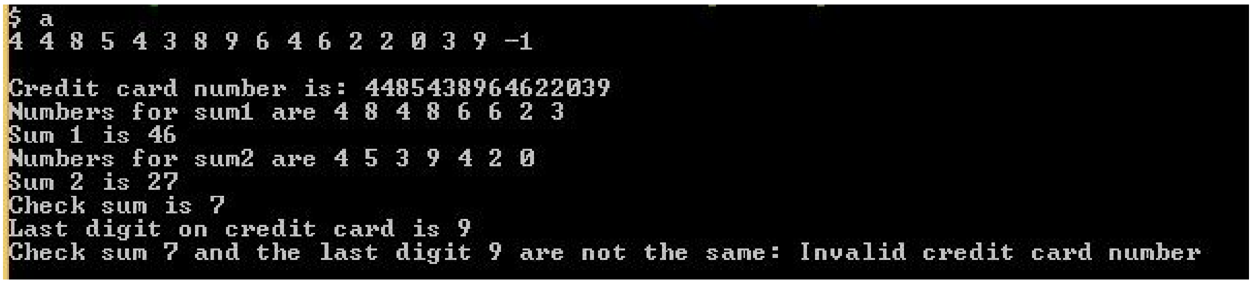


```
$ a
7 9 9 2 7 3 9 8 7 1 3
-1

Credit card number is: 79927398713
Numbers for sum1 are 9 2 3 8 1
Sum 1 is 28
Numbers for sum2 are 7 9 7 9 7
Sum 2 is 39
Check sum is 3
Last digit on credit card is 3
check sum 3 and  the last digit 3 are the same: Valid credit card number
```

eg 3:



```
$ a
4 4 8 5 4 3 8 9 6 4 6 2 2 0 3 7 -1

Credit card number is: 4485438964622037
Numbers for sum1 are 4 8 4 8 6 6 2 3
Sum 1 is 46
Numbers for sum2 are 4 5 3 9 4 2 0
Sum 2 is 27
Check sum is 7
Last digit on credit card is 7
check sum 7 and  the last digit 7 are the same: Valid credit card number
```



```
$ a
4 4 8 5 4 3 8 9 6 4 6 2 2 0 3 9 -1

Credit card number is: 4485438964622039
Numbers for sum1 are 4 8 4 8 6 6 2 3
Sum 1 is 46
Numbers for sum2 are 4 5 3 9 4 2 0
Sum 2 is 27
Check sum is 7
Last digit on credit card is 9
Check sum 7 and the last digit 9 are not the same: Invalid credit card number
```

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: Wed 10.30-11.30, Thu 1.30-4.30 pm

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