

## IPC144 S1A Assignment 2

In continuation to Assignment 1, the goal is to code procedures/functions in C programming language that prompt the customer to apply credit card application. You would need to create functions to get their data elements. These functions also responsible for checking the validity of the data elements (check the requirements listed below). Before you start the assignment, please make sure you know the followings:

- How to create a Hello-World program in C programming language.
- How to use `printf` to output characters, array of characters and strings, integer and floating-point numbers.
- How to use variables such as `char`, `int`, `float`, and array of characters.
- How to use `scanf` to read variables such as `char`, `int`, and `float`, array of characters
- How to use conditional `if-then-else` or `switch` statements in C programming language.
- How to create a function in C that return a value. How to create a function in C that perform a task without returning a value.

### Requirements:

- You need to develop an application that would allow Customers to apply a credit card. For this purpose, you need to use the data elements in the appendix in order to collect customer's information.
- When the application starts, you will ask Customer whether they are already an existing client or not
- If Customers responds with Yes, then the application would ask Card Number and password to validate the customer
- If Customer enters Card Number and password correctly, then the application pre-populates data elements as per the requirements.
- Then Customers would enter data elements as per the specs
- If Customers responds with No, then the application would get data elements from Customer as per the specs listed in the table
- Customers can try to enter Card Number and password max 3 times. After 3 times, the application displays a message and flow ends
- Customer can enter max of 3 Phone Types
- The application would check Customer's response to satisfy data requirements
- The application would display values of all data elements before ending
- Because there is no database usage, you need to hard code data elements from your code and variable would keep captured data elements.
- Create functions to get data elements

- Before you end your program, you would need to summarize all data elements and display them on monitor.
- Check the name of the customer to be valid.
  - The names should not contain any digits (e.g., “007” is not a valid name)
  - Both first name and last name should have max of 30 characters.
  - For simplicity, no middle name is needed and should not be captured.
- The customer should be 18 years and older. For simplicity, only track month and year of birth of the customer.
- Customer Phone number should be 10 character and includes only numbers from 0 to 9.
- The address should have Address, City, Province, Postal Code details
  - Length of Address1 and Address2 fields should be max 50 characters.
  - Length of City field should be max 30 characters.
  - Length of Province field should be max 2 characters and be one of the following values (AB, BC, MB, NB, NL, NT, NS, NU, ON, PE, QC, SK, YT)
  - Length of Postal code must be 6 characters.
- Process customer password by following these rules:
  - At least 8 characters long.
  - A mixture of both uppercase and lowercase letters
  - A mixture of letters and numbers
  - Inclusion of at least one special character (e.g., ! @ # ? & \$)
  - It should not include their birth year
  - It should not contain common passwords (12345678, 1234567890, qwerty12, password, qwerty123, 1q2w3e4r, 11111111, abcd1234, AAAAAAAA)
- Make sure you **create** and **use** the following functions/procedures:
  - getCardNumber
  - getPassword
  - getFirstName
  - getLastName
  - getAddress
  - getDateOfBirth
  - checkAge
  - checkName
  - checkCardNumber
  - checkPassword
  - other functions (details are provided under SPECIAL REQUIREMENTS of the appendix)

### Submission Details:

1. The assignment is **due TBD**. When due date is decided, it will be posted on Blackboard. Source code (as \*.c file) of the solution should be submitted to Blackboard. E-mail submissions are not accepted.
2. You need to convert/export your diagrams to PDF, Word or Image of the solution.
3. You should submit your solutions to Blackboard as one single file. You may want to package them as ZIP or RAR file. E-mail submissions are not accepted.
4. This assignment is composing 10% of your final mark.
5. This assignment should be done individually or with a team of 2 members. You can discuss it with your friends and classmates, but no more than one individual or team should submit the same assignment. If it is done so, all members will be punished with zero and disciplinary action will be applied. If you decide to work with a partner, both team members will get the same mark.
6. Any assignment that is submitted after due date will be punished with 10 points per day. It will be marked as zero after 2 days.
7. Please make sure all documentation is done using MS Word or PDF:
  - a. Please check the spellings, make sure your final document should have page numbers.
  - b. Please make sure you have a cover page that includes group member's name, student number and the section you enrolled. Kindly make sure to include "IPC144 S1A - Assignment 2" in the cover page.
  - c. You need to explain your design along with some assumptions, if any.

## Appendix: Specifications of Data Elements

FIELD NAME	FIELD TYPE	FIELD LENGTH	SPECIAL REQUIREMENTS	MANDATORY	PRE-POPULATED ?	IF CUSTOMER EXITS, Read Only?
First Name	Array of Characters / String / Text	30	It should not contain any digits (e.g., "007" is not a valid name)	Yes	Yes	Yes
Last Name	Array of Characters / String / Text	30	It should not contain any digits (e.g., "007" is not a valid name)	Yes	Yes	Yes
Birthdate			The customer should be 18 years and older. For simplicity, only use month and year of birth of the applicant to calculate age.	Yes		
Month	Number	2	it should be between 1 and 12			
Day	Number	2	it should be between 1 and 31			
Year	Number	4	it should be between 1900 and 2022			
Age	Number	3	Calculated		Yes	Yes
e-mail address	Array of Characters / String / Text	50	it should be valid email address with at least one @ and . Characters	Yes	Yes	No
Phone Type	Number	1	Values: 1 - Home, 2 - Work, 3 - Mobile	Yes	Yes	Yes
Phone Number	Array of Characters / String / Text		should be 10 character and includes only numbers from 0 to 9. Max 3 phone numbers can be entered	Yes	Yes	No
Home Address				Yes	Yes	Yes
Address 1	Array of Characters / String / Text	50			Yes	Yes
Address 2	Array of Characters / String / Text	50			Yes	Yes
City	Array of Characters / String / Text	30			Yes	Yes
Province	Array of Characters / String / Text	2	Length of Province field should be 2 characters and be one of the following values (AB, BC, MB, NB, NL, NT, NS, NU, ON, PE, QC, SK, YT)		Yes	Yes
Postal Code	Array of Characters / String / Text	6	Length of Postal code must be 6 characters		Yes	Yes
Living Situation	Number	1	Valid Values: 1 - Rent 2 - Own/buying 3 - Live with Parents 4 - Room or Board 5 - Other	Yes		
How long have you lived at this address?				Yes		
Years	Number	3				No
Month	Number	2	it should be between 1 and 12			No

Social Insurance Number (SIN)	Number	9	Optional, only numbers between 0 and 9	No	Yes, if exists	Yes
What kind of work do you do?				No		
Job category	Number	1	Valid Values: 1 - Category 1 2 - Category 2 3 - Category 3 4 - Category 4 5 - Category 5	No	Yes, if exists	No
Job description	Array of Characters / String / Text	30	Free Text	No	Yes, if exists	No
Yearly Income				Yes		
Personal Income	Number	8	it should be greater than \$15,000			No
Other Household income	Number	8	Optional			No
How much do you pay for housing?	Number	4		Yes	No	No
Customer ID	Number	8	It should be generated and assigned uniquely and max 8 character long			
Card Number	Array of Characters / String / Text	16	only numbers between 0 and 9			
Password	Array of Characters / String / Text	8	<ul style="list-style-type: none"> <li>• At least 8 characters long.</li> <li>• A mixture of both uppercase and lowercase letters</li> <li>• A mixture of letters and numbers</li> <li>• Inclusion of at least one special character (e.g., ! @ # ? &amp; \$)</li> <li>• It should not include their birth year</li> <li>• It should not contain common passwords (12345678, 1234567890, qwerty12, password, qwerty123, 1q2w3e4r, 11111111, abcd1234, AAAAAAAA)</li> </ul>			

Feb 2022

Yilmaz Cam

Part Time Instructor - Computer Studies  
Faculty of Continuing Education and Training  
Seneca College