Lab 01 – Design Tools

Dalhousie University

Due Date: YYYY/MMM/DD

1 Purpose

This lab is designed to familiarize you with the design tools that are required to describe and design a system.

2 Objective

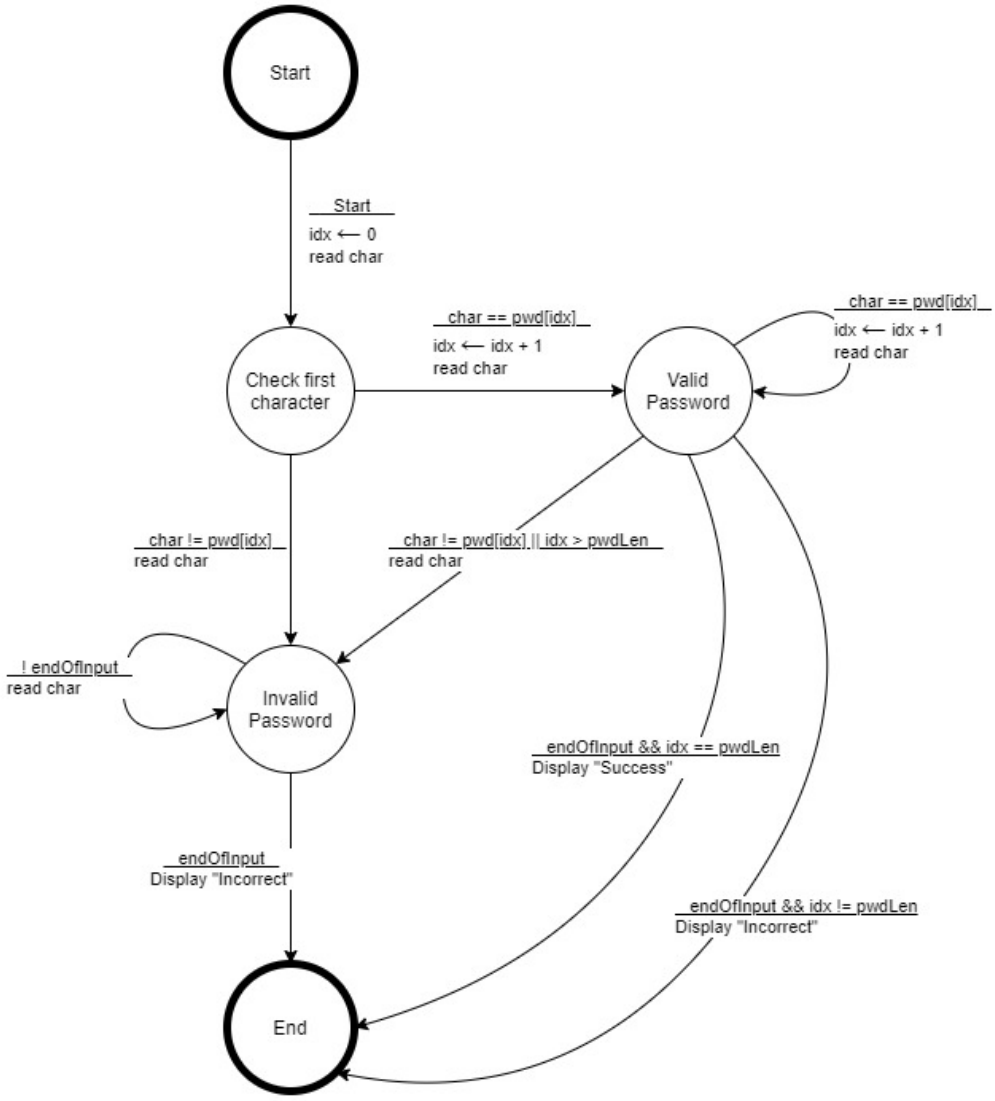
Upon completion of this lab you will have a greater understanding of the programmatic-level design tools required for describing a system. The tools that will be considered are: State Diagrams, Structured English, Data Dictionaries, and Decision Tables.

3 Background

Before diving into any problem, it is necessary to understand it. The design tools listed below are used to both help you understand the problem and explain a proposed solution.

3.1 State Diagrams

State diagrams are often used to describe finite state machines (FSM). An FSM is a machine that can be in only one of a *finite* number of states at any given time. The machine undergoes a state transition in response to external stimuli. There are two examples below. The rules for state diagrams that are to be used for this class can be found on Brightspace. Below is a state diagram showing the progression of a university student from applicant to graduate.



3.2 Structured English

Structured English is the use of the English language with the syntax of structured programming to communicate the design of a computer program to non-technical users by breaking it down into logical steps using straightforward English words[1]. The class rules for structured English can be found on Brightspace. Below is a Structured English representation of how to company formats a shipping statement. Note that the underlined terms are described in the Data Dictionary in the following section.

```
SET index to 0 *Assumes 0 indexing. Change accordingly*
READ character from input
SET state to "Check First Character"
DO WHILE state is not "End"
    READ character from input
    SELECT CASE
    CASE 1 (state = "Check First Character")
        IF character MATCHES password[ index ] THEN
            SET state to "Valid Password"
            INCREMENT index by 1
        ELSE
```

[1] https://en.wikipedia.org/wiki/Structured_English

```
            SET state to "Invalid Password"
        ENDIF
    CASE 2 (state = "Valid Password")
        IF index is GREATER THAN password length
            OR character DOESN'T MATCH password[ index ] THEN
            SET state to "Invalid Password"
        ELSE IF end of input THEN
            IF index EQUALS password length THEN
                DISPLAY "Success"
            ELSE
                DISPLAY "Incorrect"
            ENDIF
        ELSE
            INCREMENT index by 1
        ENDIF
    CASE 3 (state = "Invalid Password")
        IF end of input THEN
            DISPLAY "Incorrect"
            SET state to "End"
        ENDIF
    END-CASE
ENDDO
```

3.3 Data Dictionaries

Data dictionaries are a list of data elements and structures that are organized in a fashion providing a means to describe the data that flows throughout a given system. Simply put, they allow complex data objects to be defined in terms of simpler objects. Further description and the rules for creating a Data Dictionary can be found on Brightspace. Below is the Data Dictionary that corresponds to the Structured English example above.

```
symbol = [ ~ | ! | @ | # | $ | % | ^ | & | * | ( |
           ) | _ | + | | | ? | > | < | " | : | } |
           { | ` | [ | ] | \ | ; | ' | , | . | / ]

character = [ A-Z | a-z | 0-9 | symbol | whitespace ]

whitespace = [ *tab* | *space* | *new line* | *carriage return* ]

password =  1{character}

state = [ Check First Character | Valid Password | Invalid Password | Start |
   End ]
```

3.4 Decision Tables

Decision tables are used to enumerate all possible conditions actions. The purpose of the decision table is to connect the various conditions to their specific actions. Decision tables are often used when there are many conditions to consider when deciding on an action. The class rules for decision tables can be found on Brightspace. Below is an example of a decision table that is used to determine what catalogs should be sent to a customer. This does not tie in with the ongoing password example because that example would not be a good demonstration of a decision table. Notice that it can be reduced to the smaller table while still maintaining the same amount of information.

Conditions								
Customer ordered from Fall catalog.	Y	Y	Y	Y	N	N	N	N
Customer ordered from Christmas catalog.	Y	Y	N	N	Y	Y	N	N
Customer ordered from specialty catalog.	Y	N	Y	N	Y	N	Y	N
Actions								
Send out this year's Christmas catalog.		X		X		X		X
Send out specialty catalog.			X				X	
Send out both catalogs.	X				X			

Conditions			
Customer ordered from Fall catalog.	-	-	-
Customer ordered from Christmas catalog.	Y	-	N
Customer ordered from specialty catalog.	Y	N	Y
Actions			
Send out this year's Christmas catalog.		X	
Send out specialty catalog.			X
Send out both catalogs.	X		

4 Procedure

4.1 State Diagram

Draw a state diagram to describe an FSM for a single traffic light on a timer.

4.2 Structured English

Use Structured English to describe a binary search looking for a name in a contact list. Ensure any words that would be defined in a DD are underlined.

4.3 Data Dictionary

Create DD to describe a student at Dalhousie. The DD should include things like program, name, age, sex, etc.

4.4 Decision Table

Write a decision table to verify if an X-Makina assembler label is valid or not. The definition of a label can be found in the X-Makina assembler user guide on Brightspace.

5 Submission

This lab is to be submitted digitally (on Brightspace) before the end of lab the following week.

6 Grading

This lab will be graded for sufficient effort and completion. However, feedback will be given that will be useful for the implementation and testing portion of all assignments.

7 References

https://www.w3computing.com/systemsanalysis/writing-structured-english/
https://en.wikipedia.org/wiki/Finite-state_machine
https://brilliant.org/wiki/finite-state-machines/
J.F. Kurose, K.W. Ross, Computer Networking – A top down approach
https://www.w3computing.com/systemsanalysis/decision-table-example/