Chase Verbout
CS162
Program 4 Algorithm

This program will help a user summarize interesting syntax that was learned during the term. It uses a structure to group together data about concept names, examples, descriptions, difficulty level, and frequency rating. In addition to this, it uses a list class that will provide dynamic storage. The user will be able to add new syntax items, display them, search for a specific difficulty, and edit existing items.

Function 1: Welcome function
This function will serve the purpose of welcoming the user to the application by displaying a message: “Welcome to Syntax Simplified!”

Function 2: Menu function
This function will display the menu with all of the options that the user can select and prompt them to enter a corresponding character to choose one. The menu will list each option on a new line and append a sequence of alphabetic characters starting at “a” to the front of the listed option. For example, “a) Create a new syntax list”. In prompting the user it will ask: “Please enter the character of the option you would like to select: “. The function will return the user’s response so that the following function can decipher what they want.

Function 3: Translate function
This function will take the input from the menu response and translate it into the correct function by checking if the character matches and then calling that function. If the user selects the quit option then this function will return a signifier of that, otherwise it will return a signifier that the user would like to continue.

Function 4: Construct new syntax list
This function will allow the user to create a new list to hold their syntax information. It will prompt them to input the number of syntax items they plan to store in the list. “How many syntax items will you store: “. The input number will allow the array to be created dynamically at runtime.

Function 5: Add new syntax
This function will allow the user to enter in a new syntax if there are less than the chosen array size already stored. If not it will display a message to let the user know that there are already max syntax items. The function will prompt the user to enter the concept name, syntax example, a description, difficulty level, and the frequency rating. For example, “Please enter the concept name: “. Once the user is done entering each section, they will return to the menu.

Function 6: Display syntax
This function will display every syntax that has been entered so far. Below each syntax, their name, an example, a description, difficulty level, and frequency rating will be displayed. Once displayed, the user will be returned to the menu.

Function 7: Display syntax of a specific difficulty
This function will prompt the user to enter a difficulty level. Then the function will search through the existing syntax for a matching level. If the level is there, the syntax will be displayed. If the level is not there then a message informing the user that there is no matching entry will be displayed. The user will then be returned to the menu.

Function 8: Edit syntax item
This function will prompt the user to enter a syntax concept name. Then the function will search through the list for a matching concept. If the concept exists the user will then be asked if they would like to edit each section. If the concept is not in the list the user will be informed and receive a message will urge them to add it.

Function 9: Destruct syntax list
This function will deallocate the memory used by a syntax list when we are done with it.

Function 10: Thank you function
The last function call of the program that will display when the user decides to quit. The function will display a thank you message to the user: “Thank you for using this application!”.