Assignment #2 – Process Automation

Group members :

1. **Eda Dana(leader) 20210808072**
2. **Dilan Güler 20210808022**

General Overview of the Script:

This code is an automation script that automatically runs the CSE303Keygen application, performs specific actions, and saves the results. The code reads serial numbers from a file, clicks the corresponding buttons for each digit in the serial number, completes the process, and then saves the results to another file.

Commands Used in the Code and Their Functions:

- **#SingleInstance Force**:
  Ensures that only one instance of the script can run at a time. If the script is already running, it forces the current instance to continue and prevents new instances from starting.

- **Run("CSE303Keygen.exe")**
  Launches the specified application (CSE303Keygen.exe).

- **WinWaitActive("CSE303Keygen")**
  Waits for the application to become active.

- **Map()**
  Creates a dictionary that maps digits to their corresponding button classes.

- **FileExist("serials.txt")**
  Checks if the file serials.txt exists.

- **ProcessClose("CSE303Keygen.exe")**
  Closes the specified application.

- **StrLen(serialCleaned)**: Returns the length (number of characters) of the serialCleaned string.

- **ExitApp**
  Terminates the script.

- **FileDelete("out-group17.txt")**
  Deletes the file out-group17.txt if it exists.

- **FileRead("serials.txt")**
  Reads the content of the file serials.txt.
- **StrSplit()**
  Splits a string (e.g., serial numbers) into lines or based on a specific delimiter.
- **Trim()**
  Removes leading and trailing spaces from a string.
- **RegExReplace(serial, "[^0-9]")**
  Removes all non-numeric characters from the string.
- **ControlClick(buttonControls[digit], "CSE303Keygen")**
  Clicks a button specified by the class name within the application.
- **Sleep(900)**
  Pauses the script for 900 milliseconds.
- **ControlGetText(keyEdit, "CSE303Keygen")**
  Retrieves the text from an edit box in the application.
- **Continue**:
  Skips the remaining commands in the current loop iteration and moves to the next iteration.
- **FileAppend(result "n", "out-group17.txt")**
  Appends the content of the result variable to the file out-group17.txt.
- **ControlClick("Button1", "ahk_class #32770")** Clicks the button named Button1 in the active window.
- **WinExist("ahk_class #32770")**:
  Checks if a window with the specified class (#32770) exists and returns its ID if found.
- **WinClose("ahk_class #32770")** Closes the specified window.

Explanation of the Script:

```
; 1. Launch the Application
Run("CSE303Keygen.exe")
WinWaitActive("CSE303Keygen")  ; Wait for the application to open

```

- **Run("CSE303Keygen.exe")**: This command launches the specified application (CSE303Keygen.exe).
- **WinWaitActive("CSE303Keygen")**: This command waits for the main window of the launched application to become active.

```
; 2. Define the buttonControls Map
buttonControls := Map(
    "0", "WindowsForms10.BUTTON.app.0.141b42a_r8_ad13",
    "1", "WindowsForms10.BUTTON.app.0.141b42a_r8_ad112",
    "2", "WindowsForms10.BUTTON.app.0.141b42a_r8_ad111",
    "3", "WindowsForms10.BUTTON.app.0.141b42a_r8_ad110",
    "4", "WindowsForms10.BUTTON.app.0.141b42a_r8_ad19",
    "5", "WindowsForms10.BUTTON.app.0.141b42a_r8_ad18",
    "6", "WindowsForms10.BUTTON.app.0.141b42a_r8_ad17",
    "7", "WindowsForms10.BUTTON.app.0.141b42a_r8_ad16",
    "8", "WindowsForms10.BUTTON.app.0.141b42a_r8_ad15",
    "9", "WindowsForms10.BUTTON.app.0.141b42a_r8_ad14"
)

generateButton := "WindowsForms10.BUTTON.app.0.141b42a_r8_ad12"
clearDataButton :="WindowsForms10.BUTTON.app.0.141b42a_r8_ad11"
keyEdit := "WindowsForms10.EDIT.app.0.141b42a_r8_ad11"

```

- In this section, a **Map** called **buttonControls** is created, which contains keyvalue pairs. Each key (digits 0-9) corresponds to the ClassNN value of a specific **WindowsForms10.BUTTON.app** element. This stores the necessary information for the program to simulate pressing specific buttons.
- Other ClassNN values are assigned to specified variables (**generateButton**, **clearDataButton**, **keyEdit**).

```
if !FileExist("serials.txt") {

    ; Close the running application
    ProcessClose("CSE303Keygen.exe")  ; Enter the correct application name

    ; Terminate the script
    ExitApp
}
```

- This command checks whether the serials.txt file exists. If the file is not found, it closes the CSE303Keygen.exe application and terminates the script.

```
35
36  If (FileExist("out-group17.txt")) {
37      FileDelete("out-group17.txt")  ; Delete the file
38  }
```

- If the output file already exists, it deletes that file.

```
39
40  serials := StrSplit(FileRead("serials.txt"), "`n")
41
```

- It reads each line from the serials.txt file and splits them into an array (serials).

```
42  For serial in serials {
43      serial := Trim(serial) ; Clean leading/trailing spaces in the serial
44      If (serial = "") {
45          Continue  ; Skip if there is an empty line
46      }
47
48          serialCleaned := RegExReplace(serial, "[^0-9]") ; Retain only numbers
49          If (serialCleaned = "") {
50             Continue  ; Skip if the serial is empty
51          }
52
53
```

- It removes any spaces from each serial number.
- All letters and symbols are removed to obtain a serial consisting only of numbers. If the cleaned serial is empty, the script moves on to the next serial.

```
55      ; Split the serial and press the button for each digit
56      For i, digit in StrSplit(serialCleaned) {
57
58
59          ; Check if it is a valid digit
60          If (buttonControls[digit] = "") {  ; If there is no key
61
62              Continue
63          }
64          ; Press the corresponding digit
65          ControlClick(buttonControls[digit], "CSE303Keygen")
66          Sleep(900)  ; Wait for a short period
67      }
68
```

- For each digit of the cleaned serial, the corresponding button (from **buttonControls)** is clicked. If an invalid digit is found, an error message is displayed to the user. A 900 ms delay is added between each click.

```
69      If (StrLen(serialCleaned) <= 6) {
70      ; Click the Generate button
71      ControlClick(generateButton, "CSE303Keygen")
72      Sleep(500)  ; Wait for the Generate operation
73
```

- If the length of the serial number is 6 or shorter, the "Generate" button is clicked, and the script waits for 500 ms to allow the operation to complete.

```
76      If (StrLen(serialCleaned) < 6) {
77          ; Click the OK button and wait for the dialog box
78          WinWaitActive("ahk_class #32770")  ; Wait for the window with the OK button to open
79
80          Sleep(3000)  ; Wait for 3 seconds to view the content
81
82          ControlClick("Button1", "ahk_class #32770") ; The ClassNN value of the OK button
83          Sleep(1000)  ; Wait for the operation
84
```

- If the length of the serial number is shorter than 6, the "OK" button in the warning window is clicked by the user.

```
85          ; Manually close if the window does not close
86      If WinExist("ahk_class #32770") {
87          WinClose("ahk_class #32770")
88          Sleep(1000)  ; Wait for the closing process
89      }
90
```

- If the warning window is still open, it is closed.

```
91          ; Press the ClearData button
92          ControlClick(clearDataButton, "CSE303Keygen")
93          Sleep(500)  ; Wait for the clearing process
94          Continue
95      }
96      }
```

- The **"ClearData"** button is clicked, followed by a 500 ms wait, and then the script continues (**Continue**).

```
97          ; Read the KEY result
98          result := ControlGetText(keyEdit, "CSE303Keygen")  ; Read the key field
99
100     ; Append the result to the file (only valid results)
101 If (result != "") {
102     FileAppend(result "`n", "out-group17.txt")
103 }
104
105     ControlClick(clearDataButton, "CSE303Keygen")  ; Press the ClearData button
106     Sleep(500)  ; Wait for the clearing process
107 }
```

- The generated key (result), if not empty, creates the **out-group17.txt** file and appends the result and then clear the data.

```
109 ; 5. Close the Application
110 ProcessClose("CSE303Keygen.exe")
111
```

- Finally, the **CSE303Keygen.exe** application is closed.

Conclusion

This code is designed as an automation script that processes serial numbers. Its primary purpose is to input data into an application, perform operations, and save the results. The code is carefully structured for automation and error handling.