EXPERIMENT-7

AIM:-Apply WinRunner testing tool implementation in any real time applications.

After installing the WinRunner on your computer, invoke the WinRunner application:

- Start -> Programs ->WinRunner ->WinRunner

The opening screen of the WinRunner application is displayed, prompting you to select one of the three options:

- New Test: To create a new test script
- Open Test: To open an existing test script
- Quick Preview: To view the quick preview of WinRunner

Recording Test Cases

To test any application, first you can run the application and understand its operation. Then, you can invoke WinRunner, again run the application and record the GUI operations. During the recording mode, WinRunner will capture all your actions, which button you pressed, where you clicked the mouse etc. You need to work with the application as usual and perform all the actions to be tested. Once the recording is completed, WinRunner generates a script in TSL (Test Script Language). You can run this test script generated by WinRunner to view the results. The test results will show whether the test has passed or failed.

There are two modes of recording:

1. Context Sensitive mode: This mode of recording is used when the location of the GUI controls (i.e. X and Y coordinates) or the mouse positions are not necessary.
2. Analog mode: This mode of recording is used when the mouse positions, the location of the controls in the application, also play an important role in testing the application. This mode of recording has to be used to validate bitmaps, testing the signature etc.

The procedure for recording a test case is as follows:

Step 1: Open a new document: File -> New (or) Select "New Test" from the WinRunner's Welcome screen.

Step 2: Open (run) the application to be tested.

Step 3: Start recording a test case.
Create ->Record - Context Sensitive (or) click on the toolbar's "Record" button once, to record in Context Sensitive mode.

Step 4: Select the application to be tested by clicking on the application's title bar.

Step 5: Perform all the actions to be recorded.

Step 6: Once all required actions are recorded, stop the recording.
Create -> Stop (or) Click on the toolbar's "Stop" button to stop the recording
WinRunner generates the script for the recoded actions.

There are two modes for generating the test cases: "Global GUI map file mode" and "GUI map file per test mode". By default, it is in "Global GUI map file mode".

- In Global GUI map file mode, you have to explicitly save the information learnt by
  WinRunner. WinRunner saves it in a file with extension "gui".
  When you have to run a test, you need to load the corresponding GUI map file; otherwise it will not be able to recognize the objects in the test case and displays an error message.
- In GUI map file per test mode, WinRunner automatically saves the information it
  has learnt from the application.
  It is always preferred to work in Global GUI map file mode.

The procedure for saving the GUI map file in Global GUI map file mode is as follows:

Step 1: Record a test case by following the preceding procedure.

Step 2: Open the GUI Map Editor window as shown in Fig. Tools -> GUI Map Editor

Step 3: On selecting the GUI Map Editor.The screen as shown in figure is displayed

Step 4: Save the GUI Map file.
File -> Save As
A File dialog appears and you need to enter the filename.

Step 5: Close the GUI Map Editor window.

The procedure for loading the GUI map file is as follows:

Step 1: Open the GUI Map Editor.
Tools -> GUI Map Editor

Step 2: Close all the opened GUI Map files
File -> Close all.

The procedure for running a test case is as follows:

Step 1: Open the test script to be executed.

Step 2: Run the test
Run -> Run from top (or) press F5 to run the test.

WinRunner executes the generated script and displays the results in the Test Results window.We will now illustrate using WinRunner to test the "Standard Calculator" application available on your Windows system.You can invoke the calculator application from the desktop Start -> Programs -> Accessories -> Calculator. The GUI of the "Calculator" application is shown in Fig.

The symbols on the buttons of Calculator application represent the following functions:

+ : To perform addition - : To perform subtraction * : To perform multiplication / : To perform division . : Decimal point sqrt : To find square root of a number % : To find percent 1/x : To find inverse of a number MC : To clear the memory MR : To recall from memory MS : To save in the memory M+ : To add to the memory C :To clear the current calculation CE :To clear the displayed number +/- : To give sign to a number (positive or negative) Backspace: To remove left most digit

To test the complete functionality of the application, we need to generate test cases in such a way that all the buttons are made use of. We need to generate some test cases which will give correct output and also some test cases which will give error messages. Table gives such test cases and the expected output for each test case.

Test Cases and the Expected Output for Testing the Calculater

Test Case	Expected Output
4 1/x	0.25
- 6 sqrt	Err: "Invalid input for function"
4 C	Clears the Display
1.2 * 3	3.6
5 / 2.0	2.5
7 + 8 – 9	6
600 * 2 %	12
2, MS, C, MR	2
MC, 2, M+, 3, M+, C, MR	5

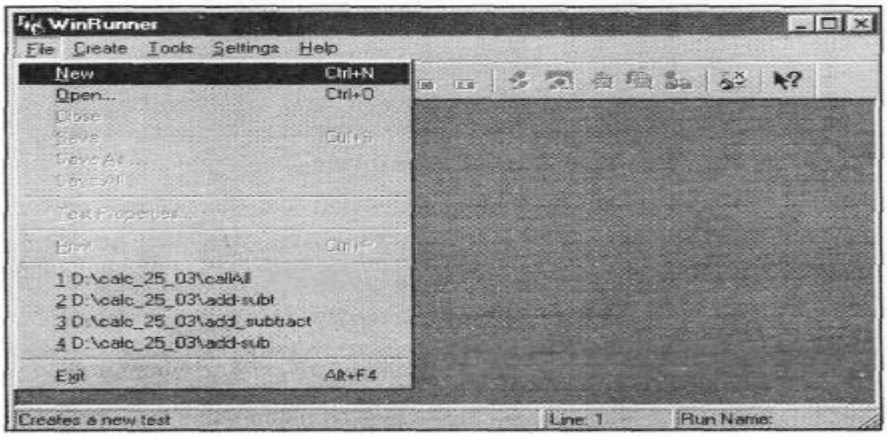
To test the functionality of the application perform the following steps:

Test Case #1: To test the Inverse operation (inverse of 4 using 1/x button)

Step 1: Open WinRunner application.

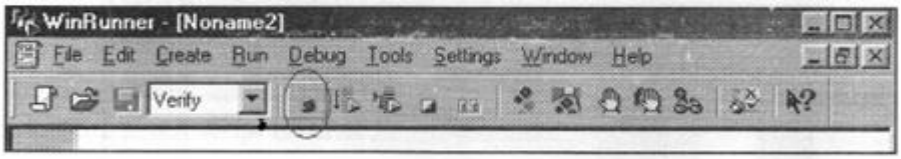
Step 2: Open Calculator application.

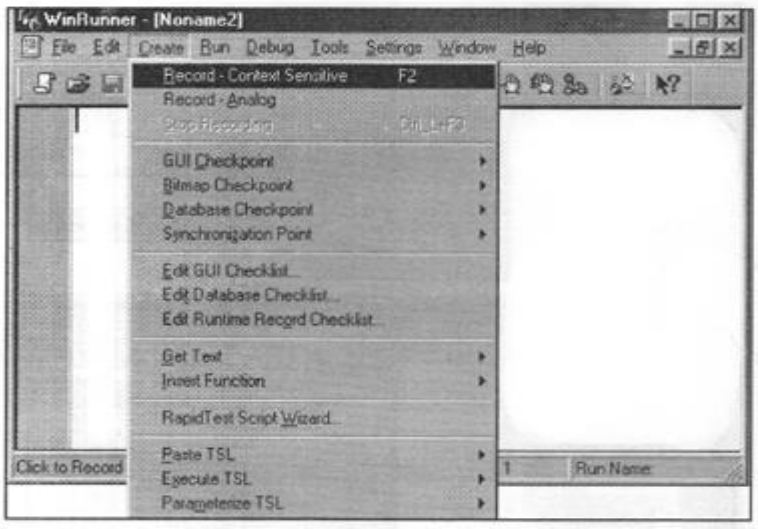
Step 3: Create a new document as shown in Figure.
File -> New or Click *Q* (New) on tool bar or press Ctrl+N



Step 4: Start recording

Create -> Record-Context Sensitive (or) press F2 (or) Click # on the toolbar Click the (Record-Context Sensitive) button on the toolbar of WinRunner as shown in Figure or Select "Record - Context sensitive" option from the "Create" menu as shown in Figure.





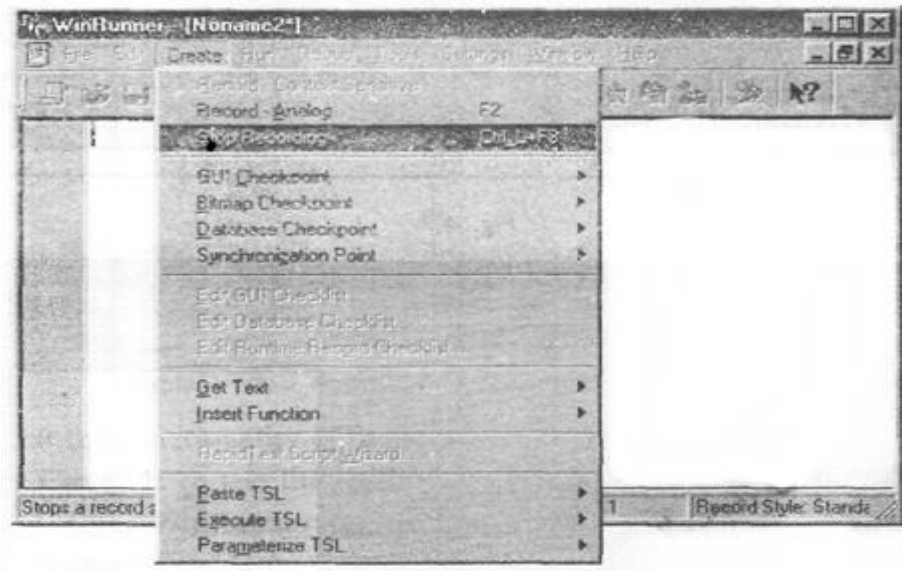
Step 5: Select the Calculator application and start recording the actions. a Click "4" on the Calculator

- Click the "1/x" button on the Calculator to find the inverse of 4.
- The result, 0.25 will be displayed on the Calculator.

Step 6: Stop the Recording process.

Create -> Stop Recording (or) Click (Stop) on toolbar

Click (Stop Recording) button on the toolbar of WinRunner as shown in Figure or Select the "Stop Recording" option from the "Create" menu as shown in Figure.



Step 8: Save the file as "inverse" in the selected folder.

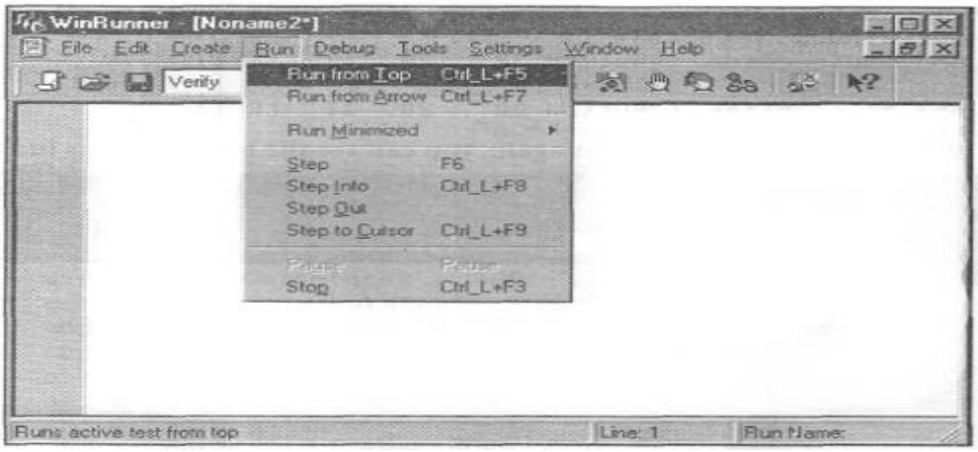
File -» Save

In the "Save" dialog box that appears, save the test script with name "inverse".

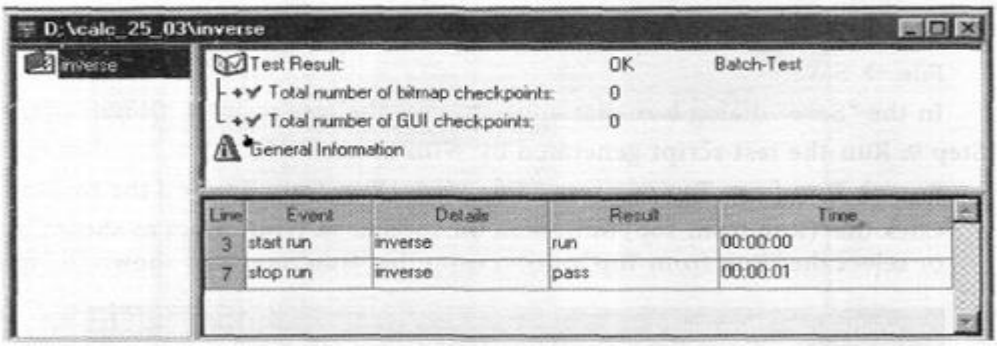
Step 9: Run the test script generated by WinRunner.

Run -> Run from Top or press F5 or Click (Run from Top) on the toolbar

Click the (Run from Top) button on the toolbar of WinRunner as shown in Figure or select the "Run from Top" option from the "Run" menu as shown in Figure.



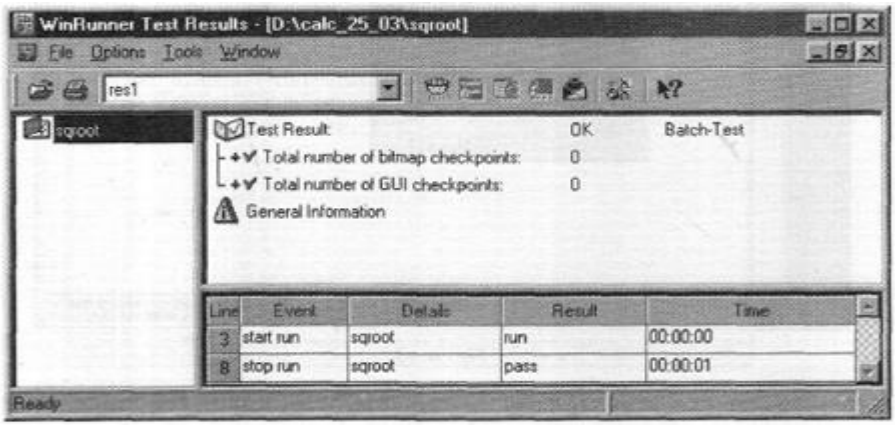
Step 10: After executing the TSL statements, WinRunner generates test results as shown in Figure. The Results column indicates whether the test has "Passed" or "Failed". The test results also give useful information such as the name of the test case, the line numbers in the test script and the time taken for executing the test case.



You can use the same procedure explained above for recording the test case. The following test script will be generated:

```
#Calculator winactivate("Calculator"); set_window("Calculator",1);
    obj_mouse_click("Button_38",20,12,LEFT);
    objmousedrag("Button_35",10,15,11,14,LEFT);
    obj_mouse_click("Button_60",20,11,LEFT);
```

When you run the test script again, you can see the test results, as in Figure.



Calling the Test Cases using "call" Function

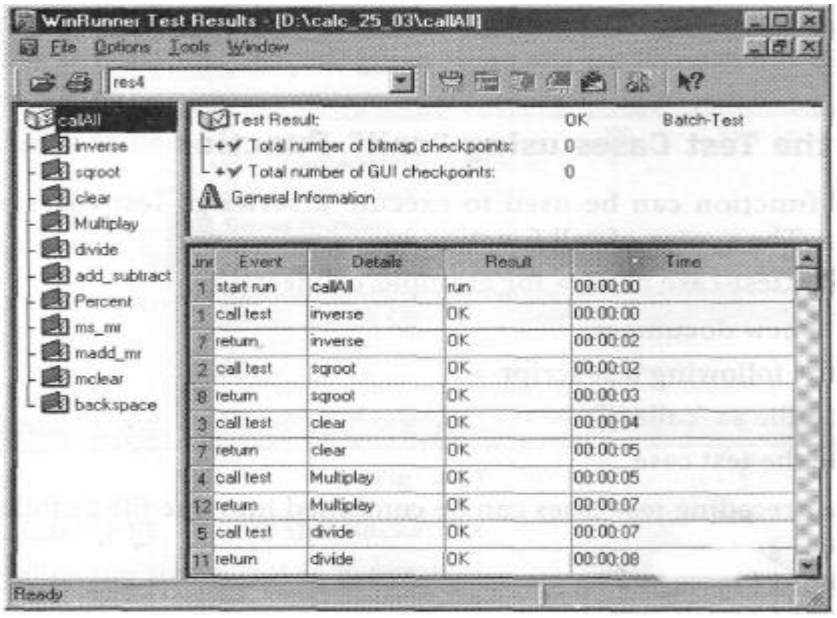
The "call" function can *oe*used to execute a series of test cases without any user interaction. The syntax of call function is:
call<test-case name> for example, call testl();

- Create a new document
- Write the following test script
- Save the file as "callAll"
- Execute the test case

All the preceding test cases can be combined into one file as follows:

call inverseO;
call sqrootO; call clearO; call MultiplayO; call divideO; call add_subtract() call PercentO; call msjnrO; call maddmrO; call mclearO; call backspaceO;

When you execute this test script, all the earlier test cases are executed in one shot. The test results screen will be as shown in Figure. As you can see from the table, the "Details" column gives the various test cases executed. The "Result" column shows whether the test has passed or failed. The "Time" column gives the time taken to execute the test case.



When you have to retest the application using the same test cases, you can run the script in unattended mode. You can save the script in a file and run the script at specified time.

This feature of WinRunner is extremely useful for regression testing. When you are developing the software, you need to run the same set of test cases many times. So, you can run the application once, generate the test script and then keep doing the regression testing. Obviously, the productivity of the test engineers will be very high when this tool is used.