

Ghidra SRE Final Challenge

Clayton B. Hodges

Colorado Mesa University

CSCI 420: Software Security

Contents

1.	Task Completion.....	2
1.1	Instructions.....	2
1.2	Puzzle.....	2
1.3	Black Box.....	6

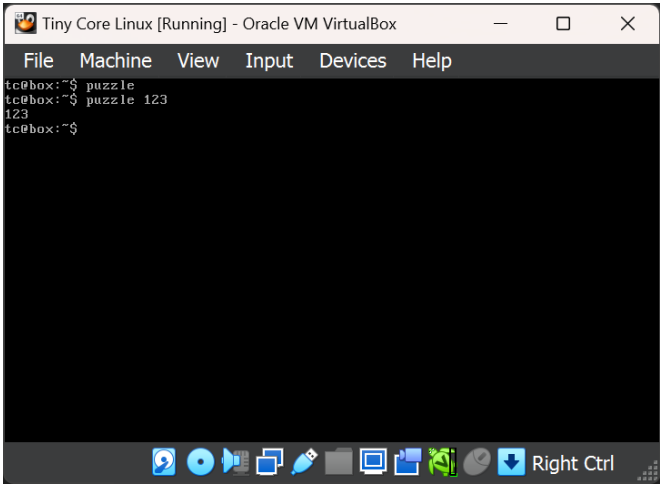
1. Task Completion

1.1 Instructions

- I. Install Ghidra, VirtualBox, and materials.
- II. Launch Tiny Core Linux and test executable(s) inside.
- III. Launch Ghidra, create a project, and upload executable(s).
- IV. Use Ghidra to deconstruct the executable(s), break down binary functions, and discover secrets.

1.2 Puzzle

- I. Entering 'puzzle' and 'puzzle 123' in Tiny Core Linux.

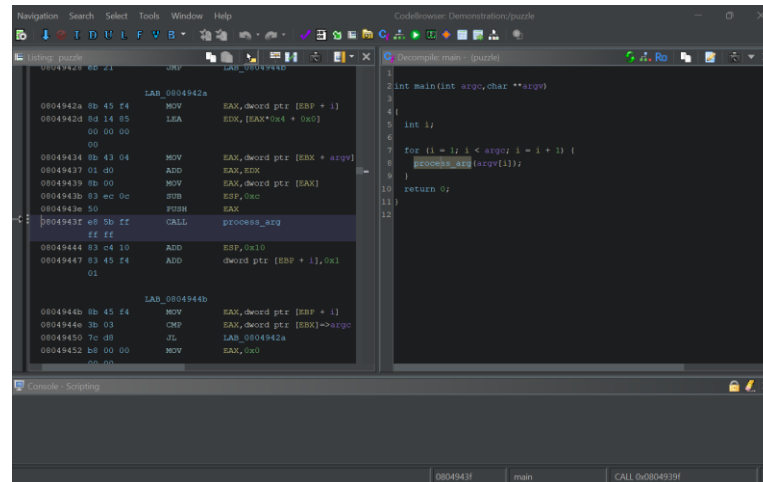


```
Tiny Core Linux [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
tc@box:~$ puzzle
tc@box:~$ puzzle 123
123
tc@box:~$
```

(Puzzle's expected output when run directly is nothing, but it prints any args it's passed.)

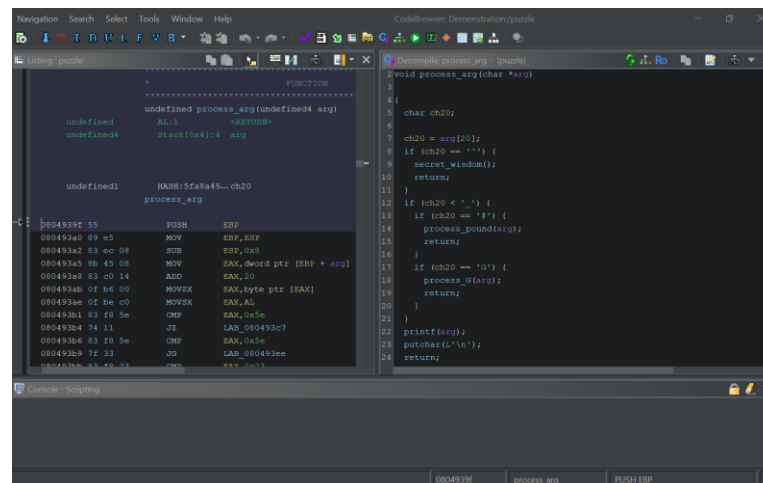
GHIDRA SRE FINAL CHALLENGE

II. Investigating Puzzle's deconstructed contents and relabeling content in Ghidra.



(The main entry point of the program calls a function we labeled process_arg.)

III. Further investigating Puzzle functions.

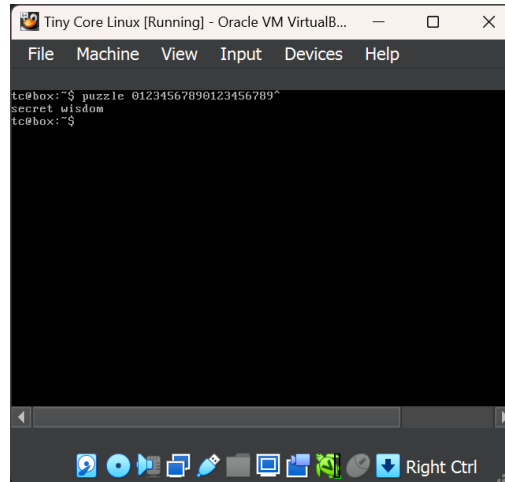


(The function(s) called by main are further examined, revealing that passing a string with '^' at

arg[20] to Puzzle will result in secret message output.)

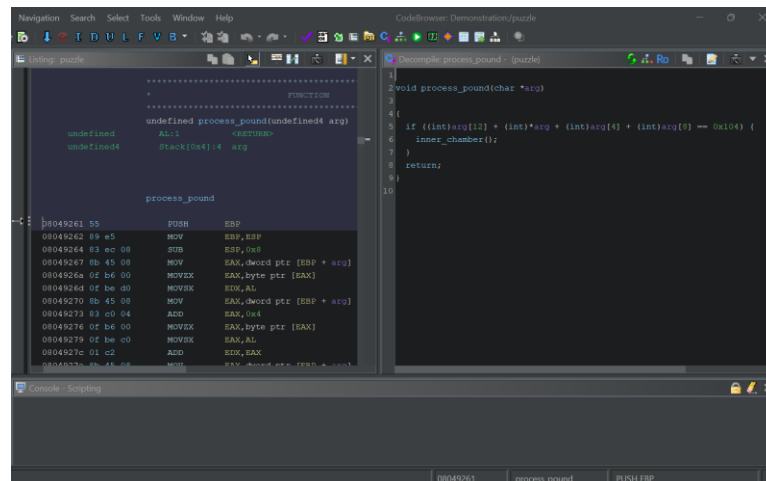
GHIDRA SRE FINAL CHALLENGE

IV. Entering 'puzzle 01234567890123456789^' in Tiny Core Linux.



(Puzzle prints 'secret wisdom' instead of the expected output.)

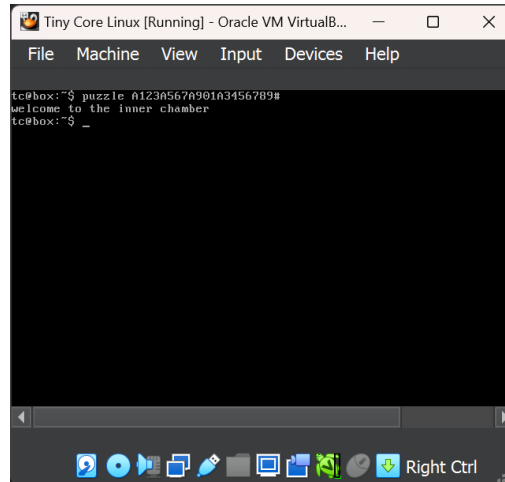
V. Further investigating Puzzle functions.



(Upon further investigation, a function reveals that passing '#' at arg[20] and ASCII values that add up to 260 at arg[0], arg[4], arg[8], and arg[12] to Puzzle will result in the output of a secret message.)

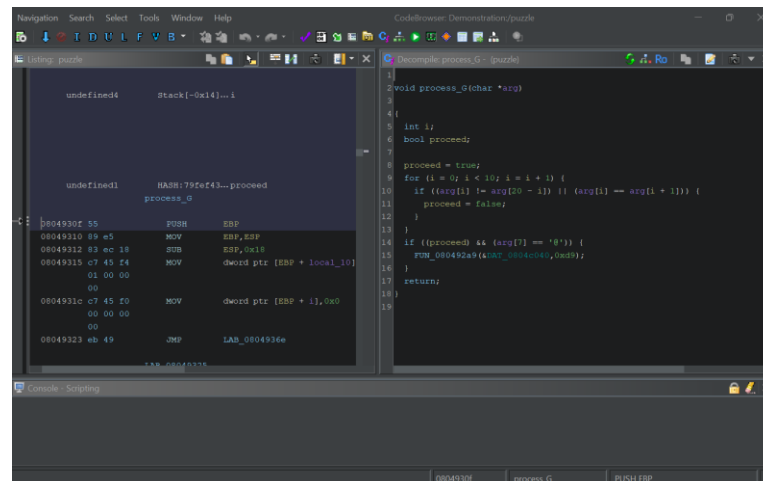
GHIDRA SRE FINAL CHALLENGE

VI. Entering 'puzzle A123A567A901A3456789#' in Tiny Core Linux.



(Puzzle prints 'welcome to the inner chamber' instead of the expected output.)

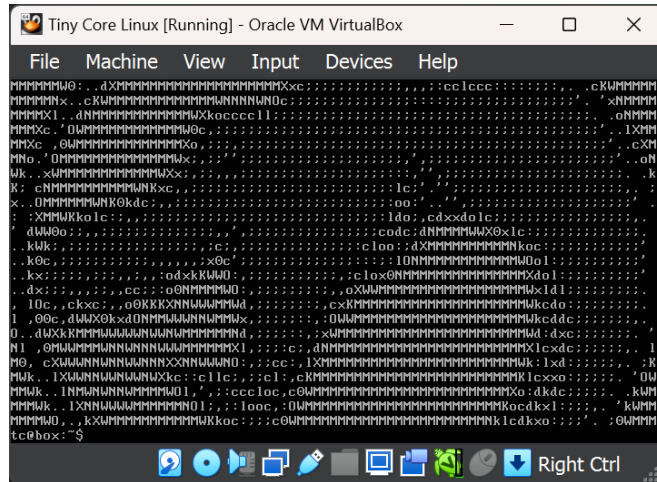
VII. Further investigating Puzzle functions.



(Upon further investigation, a function reveals that passing 'G' at arg[20] and '@' at arg[7] to Puzzle and ensuring the rest of the argument is a palindrome without repeating characters will result in the output of a secret message.)

GHIDRA SRE FINAL CHALLENGE

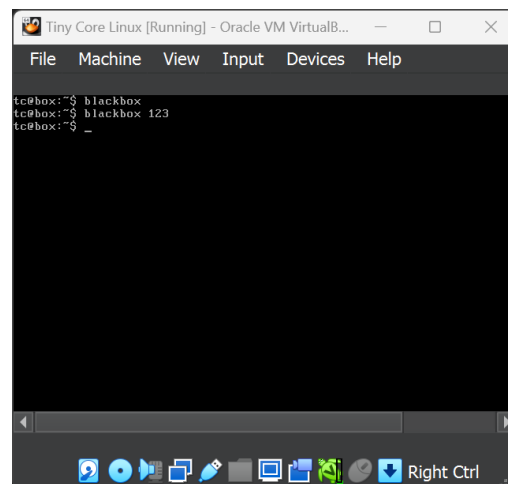
VIII. Entering 'puzzle GBABABA@ABABA@ABABAG' in Tiny Core Linux.



(Puzzle prints ASCII art of the Ghidra logo instead of the expected output.)

1.3 Black Box

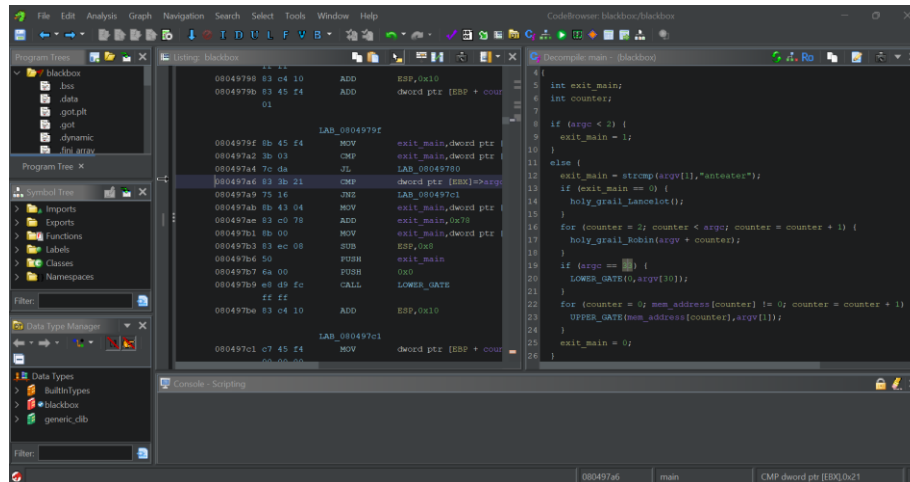
I. Entering 'blackbox' and 'blackbox 123' in Tiny Core Linux.



(Blackbox's expected output when run directly or with args is nothing.)

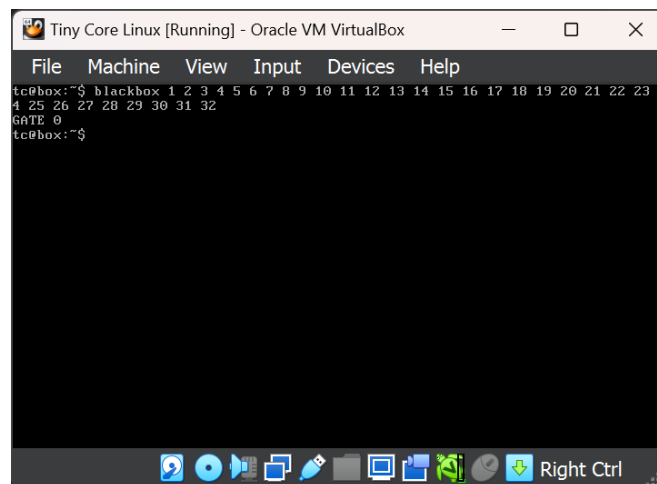
GHIDRA SRE FINAL CHALLENGE

II. Investigating Blackbox's deconstructed contents and relabeling content in Ghidra



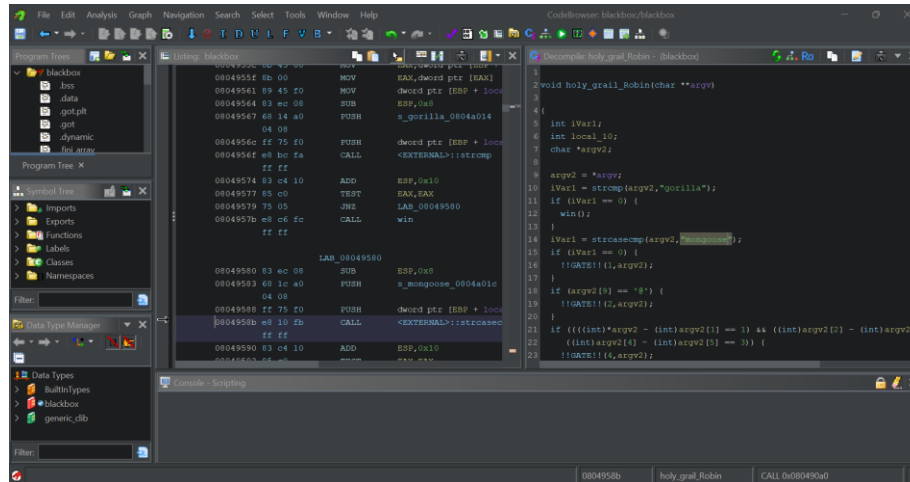
(Upon further investigation, an if check in a function will trigger gate 0 if 33 args are passed to blackbox [the function name is arg 1].)

III. Entering 'blackbox 1 2 3 ... 30 31 32' in Tiny Core Linux.



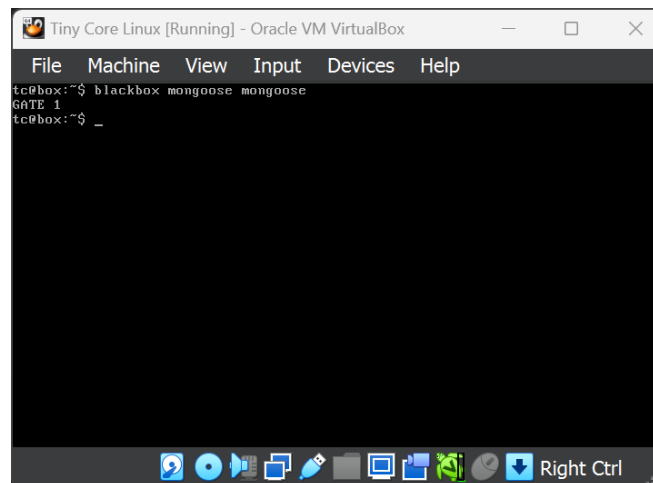
GHIDRA SRE FINAL CHALLENGE

IV. Investigating Blackbox's deconstructed contents and relabeling content in Ghidra



(Upon further investigation, an if check in a function will trigger gate 1 if the second arg passed to blackbox is 'mongoose'.)

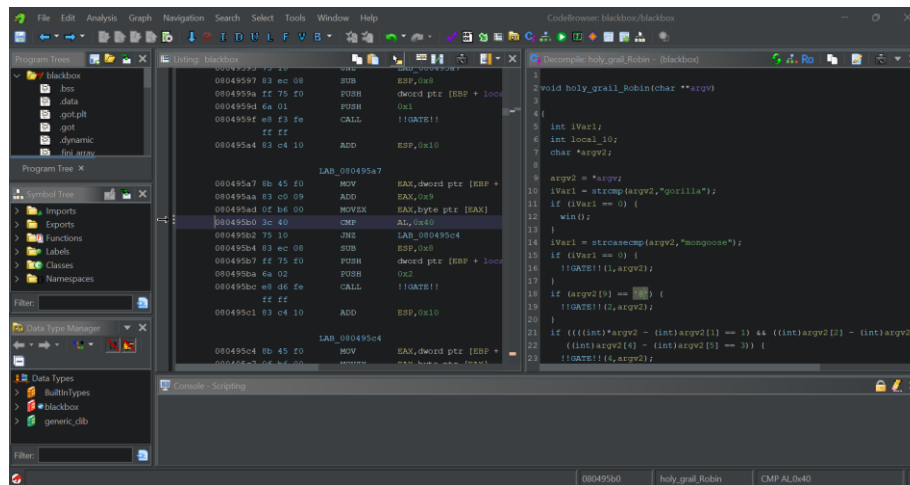
V. Entering 'blackbox mongoose mongoose' in Tiny Core Linux.



(Blackbox prints GATE 1 instead of the expected output.)

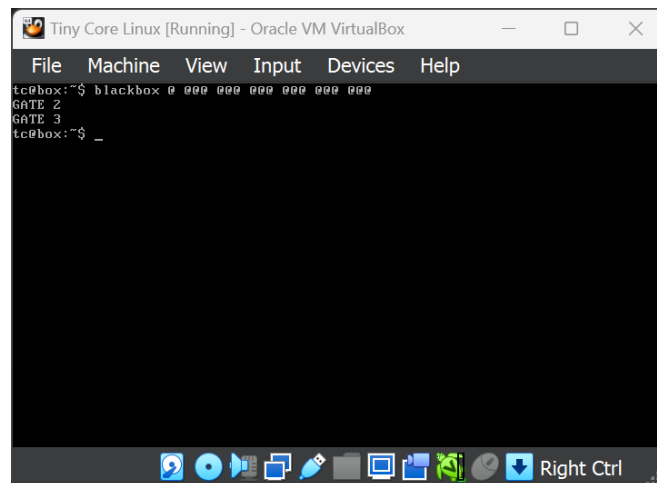
GHIDRA SRE FINAL CHALLENGE

VI. Investigating Blackbox's deconstructed contents and relabeling content in Ghidra.



(Upon further investigation, an if check in a function will trigger gate 2 if the second arg passed to blackbox has '@' at position 10.)

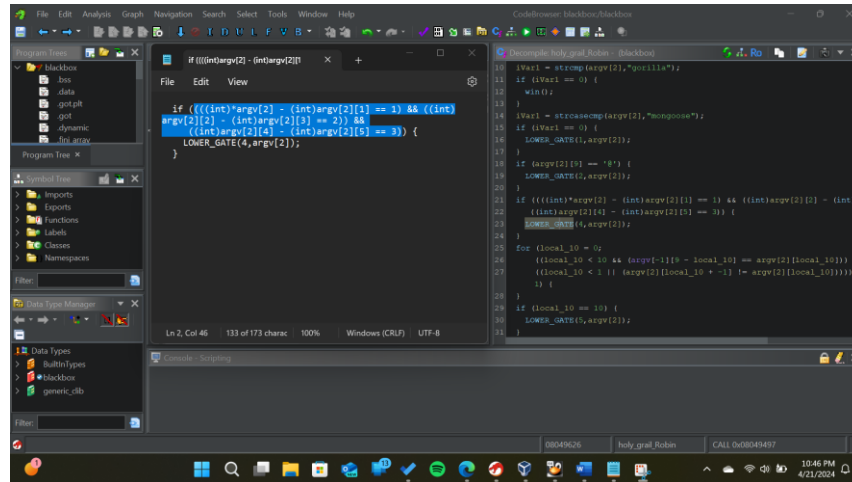
VII. Entering 'blackbox @ @@@ @@@ @@@ @@@ @@@' in Tiny Core Linux.



(Blackbox prints GATE 2 [and GATE 3 too, inadvertently] instead of the expected output.)

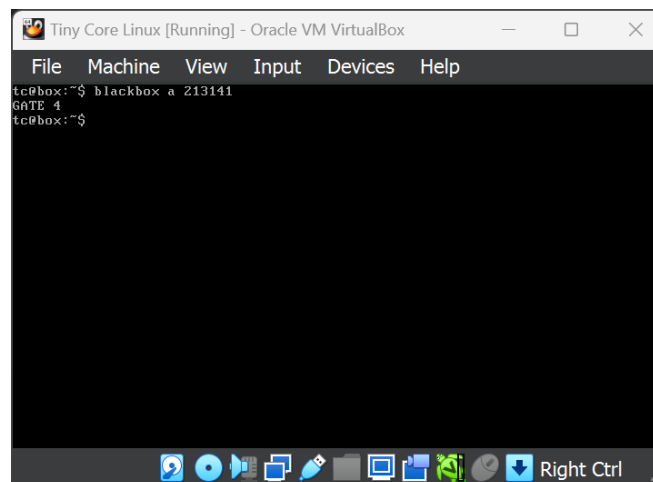
GHIDRA SRE FINAL CHALLENGE

VIII. Investigating Blackbox's deconstructed contents and relabeling content in Ghidra.



(Upon further investigation, an if check in a function will trigger gate 4 if the second arg passed to blackbox adheres to this pattern: $argv[2][0] - argv[2][1] = 1$, $argv[2][2] - argv[2][3] = 2$, and $argv[2][4] - argv[2][5] = 3$.)

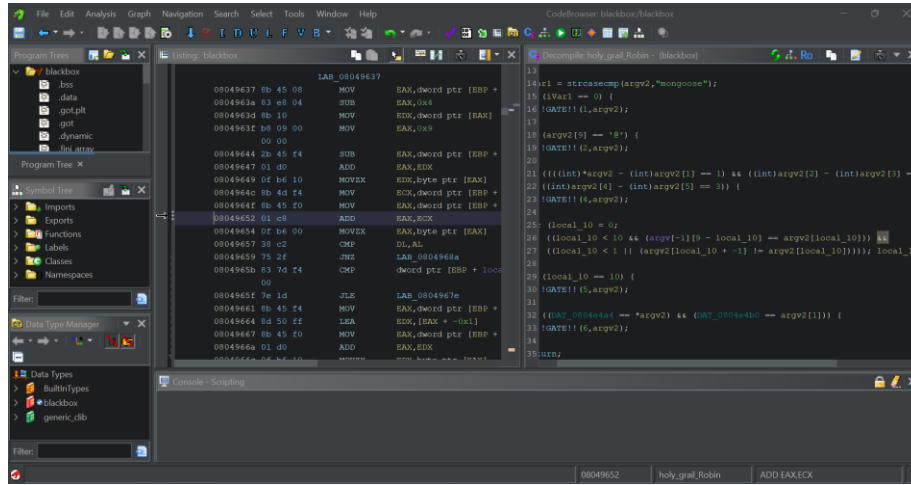
IX. Entering 'blackbox a 213141' in Tiny Core Linux.



(Blackbox prints GATE 4 instead of the expected output $2 - 1 = 1$, $3 - 1 = 2$, and $4 - 1 = 3$.)

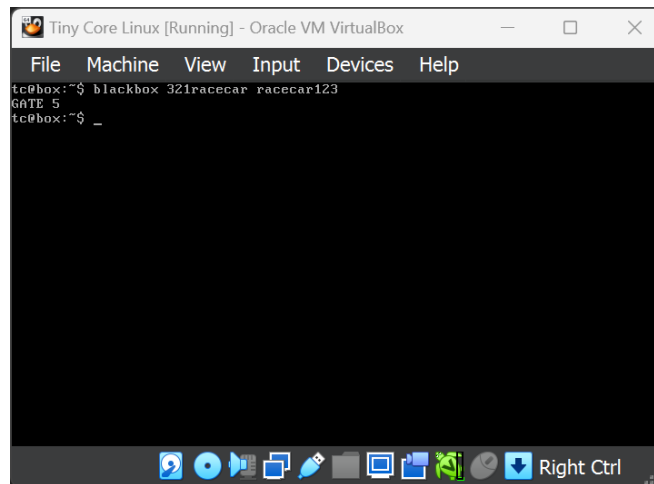
GHIDRA SRE FINAL CHALLENGE

X. Investigating Blackbox's deconstructed contents and relabeling content in Ghidra.



(Upon further investigation, an if check in a function will trigger gate 5 if the args passed to the argv[-1], the preceding arg, match arg[2], a palindrome, in reverse when passed to blackbox.)

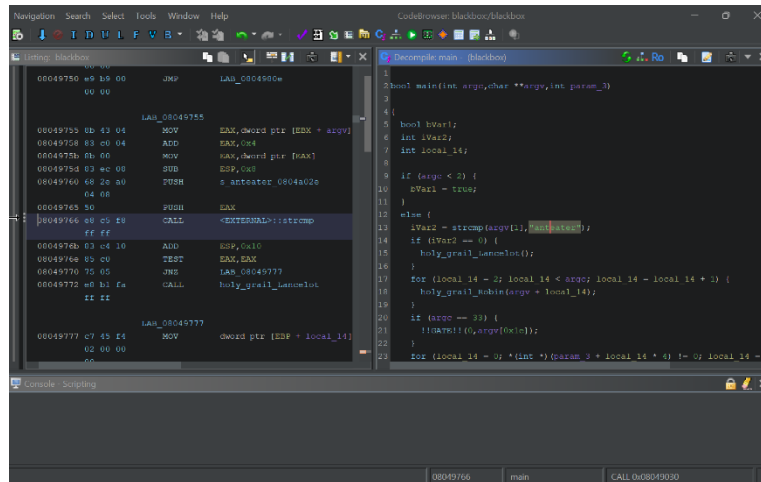
XI. Entering 'blackbox 321racecar racecar123' in Tiny Core Linux.



(Blackbox prints GATE 5 instead of the expected output.)

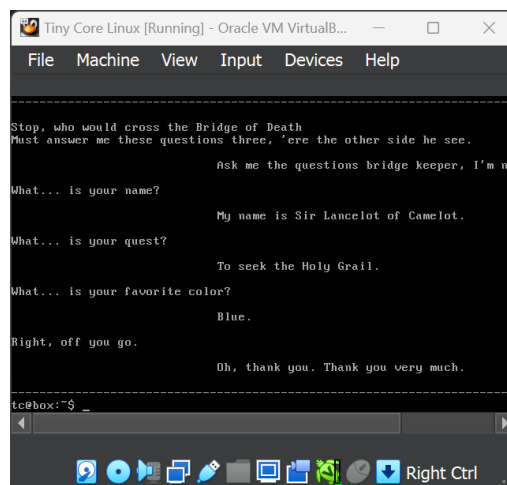
GHIDRA SRE FINAL CHALLENGE

XII. Investigating Blackbox's deconstructed contents and relabeling content in Ghidra.



(Upon further examination of the main entry point of the program's comparisons, it's revealed that passing a string with 'anteater' at arg[1] to Blackbox will result in secret message output.)

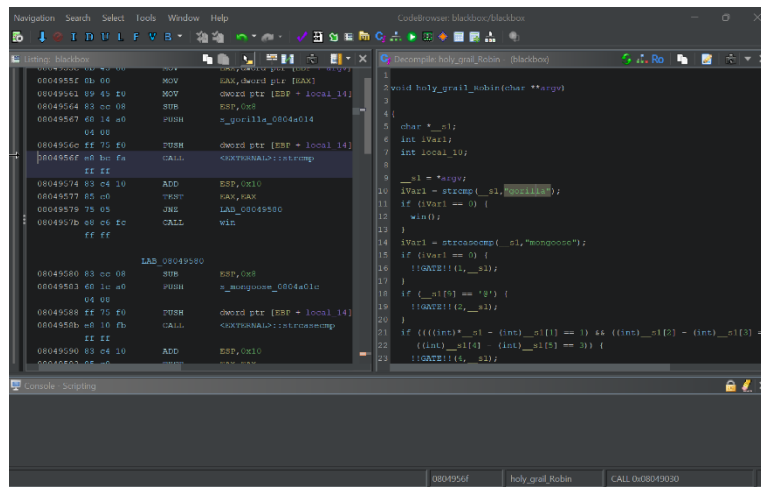
XIII. Entering 'blackbox anteater' in Tiny Core Linux.



(Blackbox prints a Monty Python dialogue instead of the expected output.)

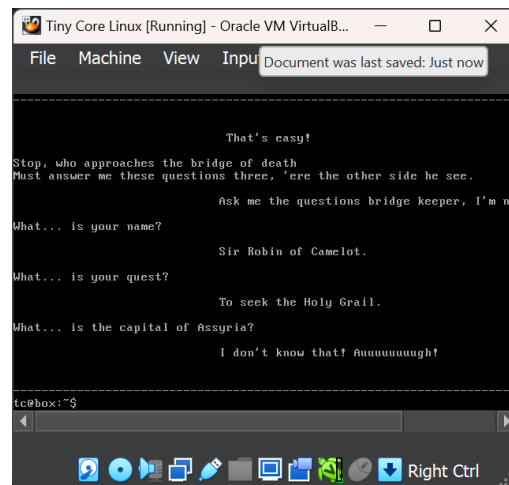
GHIDRA SRE FINAL CHALLENGE

XIV. Further investigating Blackbox functions.



(Upon further investigation, a function reveals that if arg[1] and arg[2] match as 'gorilla' when passed to Blackbox, then it will result in the output of a secret message.)

XV. Entering 'blackbox gorilla gorilla' in Tiny Core Linux.



(Blackbox prints another Monty Python dialogue instead of the expected output.)