CS250 Fall 2025 Lab 05 Lab Notebook by

Emerson Barrett, barre136

Selena Lin, lin1863

Juan Florez, jflorezg

Task 1

Idea

```
Main program:
     InstrA
     InstrB
     Call ProcX with f, g //jumps to the ProcX line
     Instr operating on r
     ...
ProcX:
     Instrs operating on f and g
     Return r //Jumps back up to Instr operating on r line
--------------------------------
LEGv8
Main program:
     InstrA
     InstrB
     Set X0-X7
     BL ProcXAddress //Jumps to Instrs operating on X0-X7 line
     Instr operating on X0-X7
     ...
ProcX:
     Instrs operating on X0-X7
     Set X0-X7
     BR LR //Jumps to Instr operating on X0-X7 line
```

Task 2

1. What do the colors of the backgrounds in the left and right half screens mean?

They correspond to which part of the code does which

2. Which labels in the C program show up in the assembly language?

The function name

3. What other parts of the C program are shown to be related to line(s) of the assembly language? (How many of these graphical linkages can you find?)

There is a return line in both, the yellow block has add and sub lines that correspond to line 7 in the C code, the parameters are set to be in registers X0-X3

4. What happens when you add the compiler flag -O for optimize?

It becomes way shorter, it only does the adding and subtracting directly without all of the loading and storing

5. What additional information is displayed when you select Output, Compile to Binary Object?

It shows memory addresses

6. Do you disagree with some portion of the response provided when for the right half screen you select + Add new... and choose Claude Explain?

Claude says the assembly is unoptimized, but it is optimized with the -O flag, and it skips initialization steps for f

Task 3

7. What do the colors of the backgrounds in the left and right half screens mean?

They correspond to which part of the code does which

8. Which labels in the C program show up in the assembly language?

The function name

9. What other parts of the C program are shown to be related to line(s) of the assembly language? (How many of these graphical linkages can you find?)

b.ge 2c <fact+0x2c> // b.tcont corresponds to the if statement, both just return in the last line

10. What happens when you add the compiler flag -O for optimize?

The code is shorter, the if-else statement gets inverted

11. What additional information is displayed when you select Output, Compile to Binary Object?

It shows memory addresses

12. Do you disagree with some portion of the response provided when for the right half screen you select + Add new... and choose Claude Explain?

No we agree, the recursive function is optimized with the -O flag.

Task 4

13. What do the colors of the backgrounds in the left and right half screens mean?

They correspond to which part of the code does which

14. Which labels in the C program show up in the assembly language?

The function name

15. What other parts of the C program are shown to be related to line(s) of the assembly language? (How many of these graphical linkages can you find?)

The yellow block determines if we move through the if block or the else block. The if block changes the return value, and the else block doesn't change it. Then when we hit the return block it returns the currently stored value.

16. What happens when you add the compiler flag -O for optimize?

It becomes way shorter, it doesn't do any storing or loading, and skips over some intermediate steps like loading and storing.

17. What additional information is displayed when you select Output, Compile to Binary Object?

It shows memory addresses

18. Do you disagree with some portion of the response provided when for the right half screen you select + Add new... and choose Claude Explain?

We agree with Claude that the function is much more optimized and primarily uses registers x0 and x1, with the final return value being stored in x0.