SI 413 Fall 2023: Homework 10

Due Monday, November 20

Your name:

Citations and collaborators:

Comments, suggestions, or questions for your instructor:

Fill out the first row of the table on a 0-5 scale before turning in.

This rubric is also available on the website under "Admin":

- **5**: Solution is completely correct, concisely presented, and neatly written.
- **4**: The solution is mostly correct, but one or two minor details were missed, or the presentation could be better.
- **3**: The main idea is correct, but there are some significant mistakes. The presentation is somewhat sloppy or confused.
- **2**: A complete effort was made, but the result is mostly incorrect.
- **1**: The beginning of an attempt was made, but the work is clearly incomplete.
- **0**: Not submitted.

Problem	1	2	Total
Self-assessment			
Final assessment			

1 LLVM IR

Consider the following function compiled into LLVM IR (or download [foo.ll])

```
target triple = "x86_64-pc-linux-gnu"

define i64 @foo(i32 %x) {
entry:
  %x.addr = alloca i32
  %res = alloca i64
  store i32 %x, ptr %x.addr
  store i64 0, ptr %res
  br label %while.cond

while.cond:
  %0 = load i32, ptr %x.addr
  %cmp = icmp sgt i32 %0, 0
  br i1 %cmp, label %while.body, label %while.end

while.body:
  %1 = load i64, ptr %res
  %mul = mul i64 %1, 10
  %add = add i64 %mul, 3
  store i64 %add, ptr %res
  %2 = load i32, ptr %x.addr
  %dec = add i32 %2, -1
  store i32 %dec, ptr %x.addr
  br label %while.cond

while.end:
  %3 = load i64, ptr %res
  ret i64 %3
}
```

a) Describe (in a sentence or two) what this function does.

b) Rewrite the foo function from the previous page so that it does not use any memory allocation, load, or store operations.

(Hint: you have to use phi instructions.)

2 Goto harmful

Write a C program with "spaghetti code" that fits the following requirements:

- It uses `goto` statements and labels
- It is relatively short (say, 30ish lines of code)
- It does *something* mildly useful or interesting
- It is hard to understand *because of the goto statements*

a) Copy your C spaghetti code program below.

b) Say what it is that your program actually does.

(Make sure part (b) is on a different page or side than part (a).)