

- for all questions, use LIR program obtained from compiling C program.

- all questions focus on the body of main. ignore allocation site variable -1

- use notation in the lectures to refer to instructions write down if indices start from 0 or 1

- in LIR function, all local variables are zero-initialized. initial state maps all var $\rightarrow 0$

extern read: () \rightarrow int;

extern print: (int) \rightarrow -;

```
fn foo(x: int)  $\rightarrow$  int {
  print(x);
  return x * 2;
}
```

main() entry:

X 1	-t2 = \$alloc 1 [-1] // ignore
X 2	c = \$copy -t2
X 3	d = \$copy c
X 4	-t3 = \$arith add 1 2 \rightarrow t3 = \$copy 3
X 5	-t4 = \$arith add -t3 3 \rightarrow t4 = \$copy 6
6	a = \$copy -t4 \rightarrow a = \$copy 6
X 7	\$store c a
X 8	-t5 = \$load d
X 9	b = \$copy -t5
10	\$jump bb1

2) constant folding & propagation optimization:

5) ded store elimination

