Topic V16

The meaning of *p

Reading: (Section 2.14)

The Meaning of *p

```
 1: void foo(int *p, int x){
      …
25:   *p = x;
      …
30:   x = *p;
31: }
```

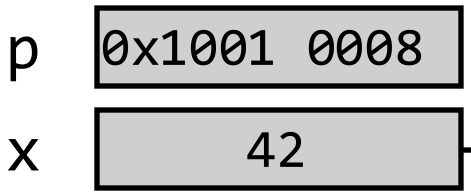
What is the meaning of the statement in line 25 of this C program?

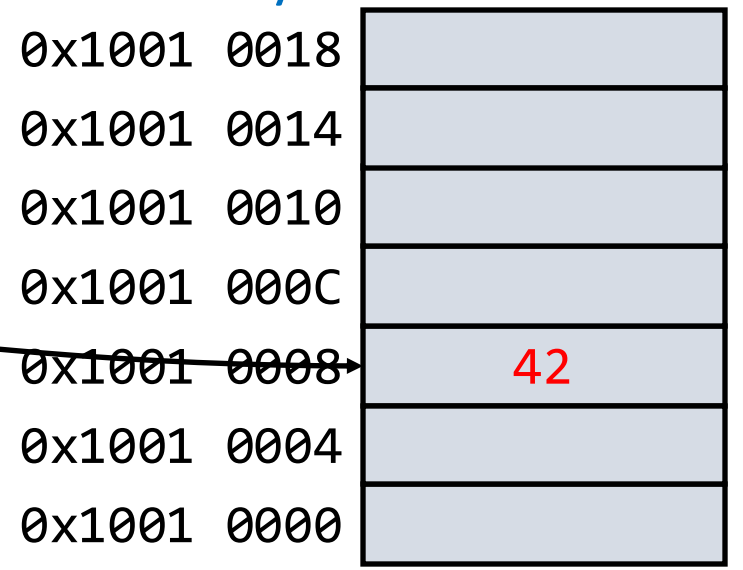
What does p contain? A memory address

What does x contain? An integer value

What action should the program do to execute line 25?

Store the value of x in the memory location whose address is in p





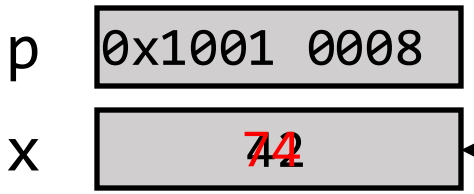
What RISC-V assembly instruction(s) is(are) needed to execute line 25?

```
sw    a1, 0(a0)  # 0(a0) ← a1
```

The Meaning of *p

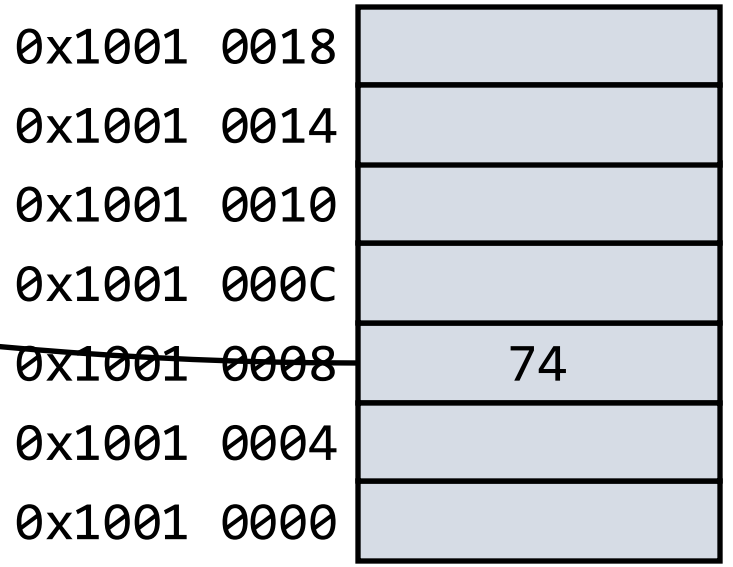
```
 1: void foo(int *p, int x){
      …
25:   *p = x;
      …
30:   x = *p;
31: }
```

What action should the program do to execute line 30? Load the value from the memory location whose address is in p



What RISC-V assembly instruction(s) is(are) needed to execute line 30?

```
lw    a1, 0(a0)  # a1 ← 0(a0)
```



The Meaning of *p

```
 1: void foo(…){
 2:   int *p;
 3:   int x;
      …
25:   *p = x;
      …
30:   x = *p;
31: }
```

Now p and x are in memory in the local stack frame.

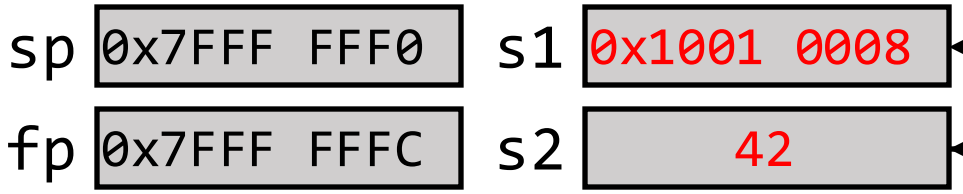
Assume that:

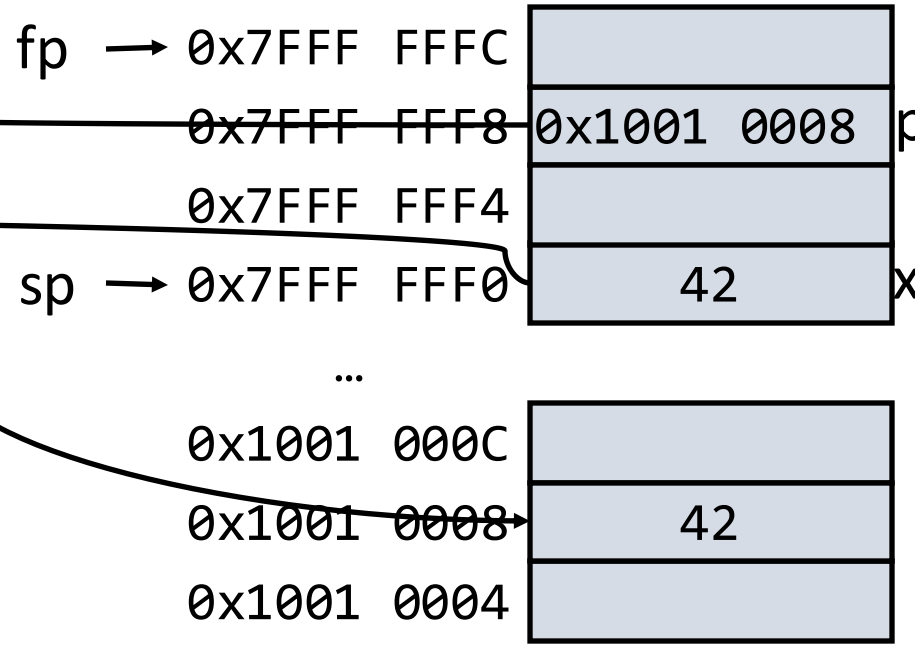
p ↔ fp – 4

x ↔ fp – 12

What actions are needed to execute line 25?

1. Load the address that is in p into a register (say s1)
2. Load the value of x into a register (say s2)
3. Store the value now in s2 into the address in s1





What RISC-V assembly instruction(s) is(are) needed to execute line 25?

```
lw    s1, -4(fp)   # s1 ← p
lw    s2, -12(fp)  # s2 ← x
sw    s2, 0(s1)    # *p ← x
```

The Meaning of *p

```
 1: void foo(…){
 2:   int *p;
 3:   int x;
      …
25:   *p = x;
      …
30:   x = *p;
31: }
```

What is the difference between the address "in p" and the address "of p"?

Now p and x are in memory in the local stack frame.

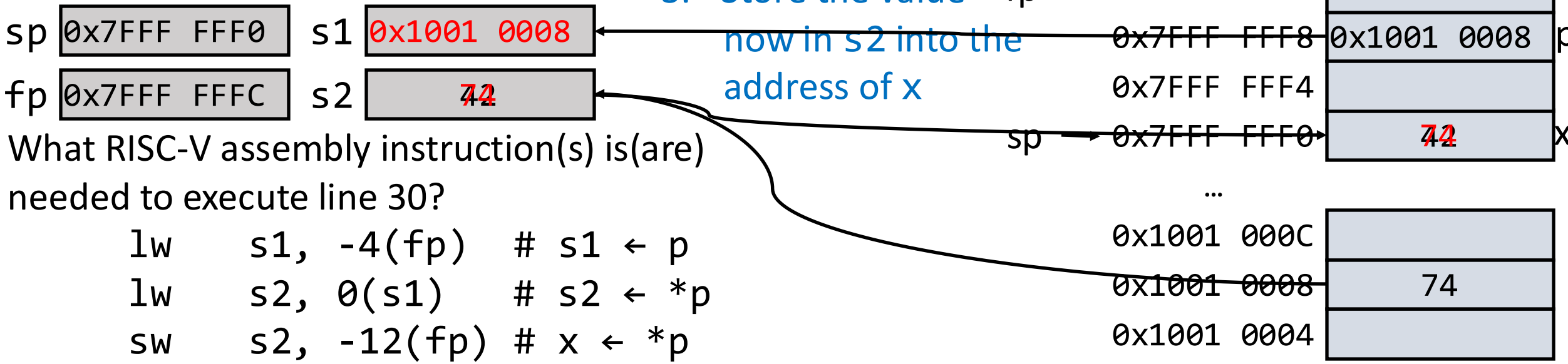
Assume that:

p ↔ fp – 4

x ↔ fp – 12

What actions are needed to execute line 30?

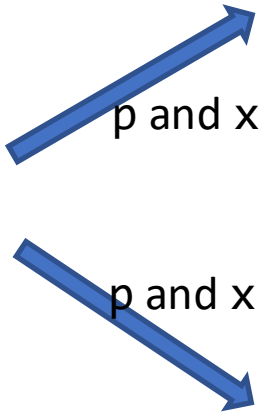
1. Load the address that is in p into a register (say s1)
2. Load the value of the memory whose address is in s1 into a register (say s2)
3. Store the value now in s2 into the address of x



What RISC-V assembly instruction(s) is(are) needed to execute line 30?

```
lw   s1, -4(fp)   # s1 ← p
lw   s2, 0(s1)    # s2 ← *p
sw   s2, -12(fp)  # x ← *p
```

```
x = *p;
```



p and x are in registers

```
lw    a1, 0(a0)  # a1 ← 0(a0)
```

p and x are in memory

```
lw    s1, -4(s0)  # s1 ← p
lw    s2, 0(s1)   # s2 ← *p
sw    s2, -12(s0) # x ← *p
```