Chapter 6 HW

6.3

6.3   a should be kept in register. It is a formal
parameter.
b should be kept in memory as a address-passing
parameter.
c should be kept in memory. Its address is used.
d and e should be kept in memory. They are
temporary variables.

6.7 a

• Step 1: Load the address of the current frame pointer into a register, say R1. (This points to the frame of show).

• Step 2: Load the static link from R1 into R1 itself. This static link points to the frame of prettyprint.

• Step 3: Add the offset to the output variable from the frame start in prettyprint, say offset_output, to R1.

• Step 4: Load the value at address R1 + offset_output into another register, say R2.

```
Plain Text
MOV R1, FP                 ; Move current frame pointer to R1
MOV R1, [R1 + STATIC_LINK] ; Follow static link to prettyprint
frame
MOV R2, [R1 + offset_output]; Move output into R2
```

6.7 b

• Step 1: Directly load the pointer from the display array at index 1 (which points to the frame of `prettyprint`) into a register, say `R1`.

• Step 2: Add the offset to the `output` variable from the start of the frame in

`prettyprint`, say `offset_output`, to `R1`.

- Step 3: Load the value at `R1 + offset_output` into another register, say `R2`.

```sql
MOV R1, DISPLAY[1]          ; Move frame pointer of prettyprint to R1
MOV R2, [R1 + offset_output]; Move output into R2
```

`prettyprint`, say `offset_output`, to `R1`.

- Step 3: Load the value at `R1 + offset_output` into another register, say `R2`.

```sql
MOV R1, DISPLAY[1]          ; Move frame pointer of prettyprint to
R1
```