$\{a \to 2, b \to 32\}$

while $a<5$ do $\{$ if $b>=64$ then $\{$ pr(a); ret; $\}$ a++; b=b*2; $\}$ $\downarrow_p$ $\{a \to 2, b \to 32\} \Rightarrow$

$\{a \to 2, b \to 32\} \downarrow_p a<5 \Downarrow$ true; $\{a \to 2, b \to 32\}$

$\{a \to 2, b \to 32\} \downarrow_p a \Downarrow 2$; $\{a \to 2, b \to 32\}$

$\{a \to 2, b \to 32\} \downarrow_p 5 \Downarrow 5$; $\{a \to 2, b \to 32\}$

$2 < 5$

$\{$ if $b>=64$ then $\{$ pr(a); ret; $\}$ a++; b=b*2; $\} \downarrow_p \{a \to 2, b \to 32\} \Rightarrow \{a \to 3, b \to 64\}$

if $b>=64$ then $\{$ pr(a); ret; $\} \downarrow_p \{a \to 2, b \to 32\} \Rightarrow \{a \to 2, b \to 32\}$

$\{a \to 2, b \to 32\} \downarrow_p b>=64 \Downarrow$ false; $\{a \to 2, b \to 32\}$

$\{a \to 2, b \to 32\} \downarrow_p b \Downarrow 32$; $\{a \to 2, b \to 32\}$;

$\{a \to 2, b \to 32\} \downarrow_p 64 \Downarrow 64$; $\{a \to 2, b \to 32\}$;

$32 < 64$

$a++ \downarrow_p \{\} :: \{a \to 2, b \to 32\} \Rightarrow \{a \to 3, b \to 32\}$

$\{\} :: \{a \to 2, b \to 32\} \downarrow_p a++ \Downarrow \{\} :: \{a \to 3, b \to 32\}$

$\{\} :: \{a \to 2, b \to 32\} \downarrow_p a \Downarrow 2$; $\{\} :: \{a \to 2, b \to 32\}$

$\{\} :: \{a \to 2, b \to 32\} \downarrow_p 1 \Downarrow 1$; $\{\} :: \{a \to 2, b \to 32\}$

$2+1=3$

upd($\{\} :: \{a \to 2, b \to 32\}$) $a\ 3 = \{\} : \{a \to 3, b \to 32\}$

$b=b*2 \downarrow_p \{\} :: \{a \to 3, b \to 32\} => \{a \to 3, b \to 64\}$

...

while $a<5$ do $\{$ if $b>=64$ then $\{$ pr(a); ret; $\}$ a++; b=b*2; $\} \downarrow_p \{a \to 3, b \to 64\} \Rightarrow$?

$\{a \to 3, b \to 64\} \downarrow_p a<5 \Downarrow$ true; $\{a \to 3, b \to 64\}$

...

$\{$ if $b>=64$ then $\{$ pr(a); ret; $\}$ a++ b=b*2; $\} \downarrow_p \{a \to 3, b \to 64\}$

if $b>=64$ then $\{$ pr(a); ret; $\} \downarrow_p \{a \to 3, b \to 64\}$

$\{a \to 3, b \to 64\} \downarrow_p b>=64 \Downarrow$ true; $\{a \to 3, b \to 64\}$

...

pr(a);

ret; (note: SEQ_EARLYRET catches this and stops after ret;)

ret; will give us a return frame, but what do we do after that? The semantics for WHILE-TRUE do not include a case for when the output is a return frame, so the program will break

```
var a = 2;
var b = 32;

while a < 5 do {
  if b >= 64 then {
    print_int(a);
    return;
  }
  a++;
  b = b * 2;
}
print_int(b);
return;
```

(pr(a))
(ret)

Shortenings for our sake and yours