

Nov 07, 16 0:17

## hw06-LET-output-trace

Page 1/10

```

1 [dshin@acacia let]$ Let -t /home/pl/hw06/tests/const-negative
2 Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.
3 > % =====
4 % negative constant. (value=-27)
5
6 -27
7
8 % =====
9 |+ exp=-27
10 || env=[x=10,v=5,i=1]
11 |- val=-27
12 -27
13 >
14 [dshin@acacia let]$ Let -t /home/pl/hw06/tests/const-positive
15 Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.
16 > % =====
17 % positive constant. (value=12)
18
19 12
20
21 % =====
22 |+ exp=12
23 || env=[x=10,v=5,i=1]
24 |- val=12
25 12
26 >
27 [dshin@acacia let]$ Let -t /home/pl/hw06/tests/diff-1
28 Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.
29 > % =====
30 % diff, where env=[i=1,v=5,x=10]. (value=9)
31
32 -(x,1)
33
34 % =====
35 |+ exp=-(x,1)
36 || env=[x=10,v=5,i=1]
37 ||+ exp=x
38 ||| env=[x=10,v=5,i=1]
39 ||- val=10
40 ||+ exp=1
41 ||| env=[x=10,v=5,i=1]
42 ||- val=1
43 |- val=9
44 9
45 >
46 [dshin@acacia let]$ Let -t /home/pl/hw06/tests/diff-2
47 Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.
48 > % =====
49 % diff, where env=[i=1,v=5,x=10]. (value=3)
50
51 -(1,-2)
52
53 % =====
54 |+ exp=-(1,-2)
55 || env=[x=10,v=5,i=1]
56 ||+ exp=1
57 ||| env=[x=10,v=5,i=1]
58 ||- val=1
59 ||+ exp=-2
60 ||| env=[x=10,v=5,i=1]
61 ||- val=-2
62 |- val=3
63 3
64 >
65 [dshin@acacia let]$ Let -t /home/pl/hw06/tests/diff-error-1
66 Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.

```

Nov 07, 16 0:17

## hw06-LET-output-trace

Page 2/10

```

67 > % =====
68 % diff, where env=[i=1,v=5,x=10]. (value=error)
69 -(zero?(0),x)
70
71 % =====
72 |+ exp=-(zero?(0),x)
73 || env=[x=10,v=5,i=1]
74 ||+ exp=zero?(0)
75 ||| env=[x=10,v=5,i=1]
76 |||+ exp=0
77 |||| env=[x=10,v=5,i=1]
78 |||- val=0
79 ||- val=#t
80 expval->num: arg=#(struct:bool-val #t)
81 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/value.scm:13:17
: (eopl:error (quote expval->num) "arg=~a" val)
82 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:37:30:
(- (expval->num (value-of expl env)) (expval->num (value-of exp2 env)))
83 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:37:21:
(num-val (- (expval->num (value-of expl env)) (expval->num (value-of exp2 env))))
84 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:23:4: (
let-values (((val) (value-of-l exp env))) (trace-run-exit val) val)
85 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:7:28: (
expval->string (run string))
86 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:7:15: (
format "~a" (expval->string (run string)))
87 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:7:6: (d
isplay (format "~a" (expval->string (run string))))
88 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:5:4: (l
et-values (((string) (readfile file))) (display string) (display (format "~a" ((
....) ....)))) (newline))
89
90
91 === context ===
92 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:28:2: v
alue-of-l
93 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:21:2: v
alue-of
94 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:4:2: ru
nfile
95
96 >
97 [dshin@acacia let]$ Let -t /home/pl/hw06/tests/diff-error-2
98 Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.
99 > % =====
100 % diff, where env=[i=1,v=5,x=10]. (value=error)
101
102 -(x,zero?(0))
103
104 % =====
105 |+ exp=-(x,zero?(0))
106 || env=[x=10,v=5,i=1]
107 ||+ exp=x
108 ||| env=[x=10,v=5,i=1]
109 ||- val=10
110 ||+ exp=zero?(0)
111 ||| env=[x=10,v=5,i=1]
112 |||+ exp=0
113 |||| env=[x=10,v=5,i=1]
114 |||- val=0
115 ||- val=#t
116 expval->num: arg=#(struct:bool-val #t)
117 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/value.scm:13:17
: (eopl:error (quote expval->num) "arg=~a" val)
118 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:37:30:

```

Nov 07, 16 0:17	hw06-LET-output-trace	Page 3/10
119	(- (expval->num (value-of exp1 env)) (expval->num (value-of exp2 env))) /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:37:21: (num-val (- (expval->num (value-of exp1 env)) (expval->num (value-of exp2 env))) ) /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:23:4: (let-values ((val) (value-of-1 exp env))) (trace-run-exit val) val) /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:7:28: (expval->string (run string)) /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:7:15: (format "~a" (expval->string (run string))) /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:7:6: (display (format "~a" (expval->string (run string)))) /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:5:4: (let-values ((string) (readfile file))) (display string) (display (format "~a" (....) ....))) (newline)) 125 126 127 === context === 128 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:28:2: value-of-1 129 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:21:2: value-of 130 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:4:2: runfile 131 132 > 133 [dshin@acacia let]\$ Let -t /home/pl/hw06/tests/diff-nested-left 134 Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc. 135 > % ===== 136 % nested diff. (-11) 137 138 -(- (44, 33), 22) 139 140 % ===== 141  + exp=-(- (44, 33), 22) 142    env=[x=10,v=5,i=1] 143   + exp=- (44, 33) 144     env=[x=10,v=5,i=1] 145    + exp=44 146      env=[x=10,v=5,i=1] 147    - val=44 148    + exp=33 149      env=[x=10,v=5,i=1] 150    - val=33 151   - val=11 152   + exp=22 153     env=[x=10,v=5,i=1] 154   - val=22 155  - val=-11 156 -11 157 > 158 [dshin@acacia let]\$ Let -t /home/pl/hw06/tests/diff-nested-right 159 Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc. 160 > % ===== 161 % nested diff. (44) 162 163 -(55, -(22, 11)) 164 165 % ===== 166  + exp=- (55, -(22, 11)) 167    env=[x=10,v=5,i=1] 168   + exp=55 169     env=[x=10,v=5,i=1] 170   - val=55 171   + exp=- (22, 11) 172     env=[x=10,v=5,i=1]	

Nov 07, 16 0:17	hw06-LET-output-trace	Page 4/10
173	+ exp=22 174      env=[x=10,v=5,i=1] 175    - val=22 176    + exp=11 177      env=[x=10,v=5,i=1] 178    - val=11 179   - val=11 180  - val=44 181 44 182 > 183 [dshin@acacia let]\$ Let -t /home/pl/hw06/tests/if-error 184 Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc. 185 > % ===== 186 % if with non bool checking. (error) 187 188 if 1 then 2 else 3 189 190 % ===== 191  + exp=if 1 then 2 else 3 192    env=[x=10,v=5,i=1] 193   + exp=1 194     env=[x=10,v=5,i=1] 195   - val=1 196 expval->bool: arg=#(struct:num-val 1) 197 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/value.scm:19:17: : (eopl:error (quote expval->bool) "arg=~a" val) 198 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:40:19: (if (expval->bool (value-of exp1 env)) (value-of exp2 env) (value-of exp3 env)) 199 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:23:4: (let-values ((val) (value-of-1 exp env))) (trace-run-exit val) val) 200 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:7:28: (expval->string (run string)) 201 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:7:15: (format "~a" (expval->string (run string))) 202 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:7:6: (display (format "~a" (expval->string (run string)))) 203 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:5:4: (let-values ((string) (readfile file))) (display string) (display (format "~a" (....) ....))) (newline)) 204 205 206 === context === 207 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:28:2: value-of-1 208 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:21:2: value-of 209 /home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:4:2: runfile 210 211 > 212 [dshin@acacia let]\$ Let -t /home/pl/hw06/tests/if-false-1 213 Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc. 214 > % ===== 215 % if conditional. (value=4) 216 217 if zero?(1) then 3 else 4 218 219 % ===== 220  + exp=if zero?(1) then 3 else 4 221    env=[x=10,v=5,i=1] 222   + exp=zero?(1) 223     env=[x=10,v=5,i=1] 224    + exp=1 225      env=[x=10,v=5,i=1] 226    - val=1 227   - val=#f	

Nov 07, 16 0:17	hw06-LET-output-trace	Page 5/10
228	+ exp=4	
229	env=[x=10,v=5,i=1]	
230	- val=4	
231	- val=4	
232	4	
233	>	
234	[dshin@acacia let]\$ Let -t /home/pl/hw06/tests/if-false-2	
235	Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.	
236	> % =====	
237	% if conditional where false exp is not evaluated. (value=3)	
238		
239	if zero?(0) then 3 else foo	
240		
241	% =====	
242	+ exp=if zero?(0) then 3 else foo	
243	env=[x=10,v=5,i=1]	
244	+ exp=zero?(0)	
245	env=[x=10,v=5,i=1]	
246	+ exp=0	
247	env=[x=10,v=5,i=1]	
248	- val=0	
249	- val=#t	
250	+ exp=3	
251	env=[x=10,v=5,i=1]	
252	- val=3	
253	- val=3	
254	3	
255	>	
256	[dshin@acacia let]\$ Let -t /home/pl/hw06/tests/if-true-1	
257	Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.	
258	> % =====	
259	% if conditional. (value=3)	
260		
261	if zero?(0) then 3 else 4	
262		
263	% =====	
264	+ exp=if zero?(0) then 3 else 4	
265	env=[x=10,v=5,i=1]	
266	+ exp=zero?(0)	
267	env=[x=10,v=5,i=1]	
268	+ exp=0	
269	env=[x=10,v=5,i=1]	
270	- val=0	
271	- val=#t	
272	+ exp=3	
273	env=[x=10,v=5,i=1]	
274	- val=3	
275	- val=3	
276	3	
277	>	
278	[dshin@acacia let]\$ Let -t /home/pl/hw06/tests/if-true-2	
279	Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.	
280	> % =====	
281	% if conditional where true exp is not evaluated. (value=3)	
282		
283	if zero?(1) then foo else 3	
284		
285	% =====	
286	+ exp=if zero?(1) then foo else 3	
287	env=[x=10,v=5,i=1]	
288	+ exp=zero?(1)	
289	env=[x=10,v=5,i=1]	
290	+ exp=1	
291	env=[x=10,v=5,i=1]	
292	- val=1	
293	- val=#f	

Nov 07, 16 0:17	hw06-LET-output-trace	Page 6/10
294	+ exp=3	
295	env=[x=10,v=5,i=1]	
296	- val=3	
297	- val=3	
298	3	
299	>	
300	[dshin@acacia let]\$ Let -t /home/pl/hw06/tests/let	
301	Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.	
302	> % =====	
303	% let. (value=3)	
304		
305	let x=3	
306	in x	
307		
308	% =====	
309	+ exp=let x=3 in x	
310	env=[x=10,v=5,i=1]	
311	+ exp=3	
312	env=[x=10,v=5,i=1]	
313	- val=3	
314	+ exp=x	
315	env=[x=3,x=10,v=5,i=1]	
316	- val=3	
317	- val=3	
318	3	
319	>	
320	[dshin@acacia let]\$ Let -t /home/pl/hw06/tests/let-body	
321	Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.	
322	> % =====	
323	% make sure the let body is evaluated. (value=7)	
324		
325	let x=8 in -(x,1)	
326		
327	% =====	
328	+ exp=let x=8 in -(x,1)	
329	env=[x=10,v=5,i=1]	
330	+ exp=8	
331	env=[x=10,v=5,i=1]	
332	- val=8	
333	+ exp=- (x,1)	
334	env=[x=8,x=10,v=5,i=1]	
335	+ exp=x	
336	env=[x=8,x=10,v=5,i=1]	
337	- val=8	
338	+ exp=1	
339	env=[x=8,x=10,v=5,i=1]	
340	- val=1	
341	- val=7	
342	- val=7	
343	7	
344	>	
345	[dshin@acacia let]\$ Let -t /home/pl/hw06/tests/let-nested	
346	Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.	
347	> % =====	
348	% nested let make lexical scoping. p66. (value=3)	
349		
350	let z=5	
351	in let x=3	
352	in let y=- (x,1)	
353	in let x=4	
354	in - (z,- (x,y))	
355		
356	% =====	
357	+ exp=let z=5 in let x=3 in let y=- (x,1) in let x=4 in - (z,- (x,y))	
358	env=[x=10,v=5,i=1]	
359	+ exp=5	

Nov 07, 16 0:17

hw06-LET-output-trace

Page 7/10

```

360 ||| env=[x=10,v=5,i=1]
361 ||- val=5
362 ||+ exp=let x=3 in let y=-(x,1) in let x=4 in -(z,-(x,y))
363 ||| env=[z=5,x=10,v=5,i=1]
364 |||+ exp=3
365 ||| env=[z=5,x=10,v=5,i=1]
366 |||- val=3
367 |||+ exp=let y=-(x,1) in let x=4 in -(z,-(x,y))
368 ||| env=[x=3,z=5,x=10,v=5,i=1]
369 |||+ exp=-(x,1)
370 ||| env=[x=3,z=5,x=10,v=5,i=1]
371 |||+ exp=x
372 ||| env=[x=3,z=5,x=10,v=5,i=1]
373 ||| val=3
374 |||+ exp=1
375 ||| env=[x=3,z=5,x=10,v=5,i=1]
376 ||| val=1
377 ||| val=2
378 |||+ exp=let x=4 in -(z,-(x,y))
379 ||| env=[y=2,x=3,z=5,x=10,v=5,i=1]
380 |||+ exp=4
381 ||| env=[y=2,x=3,z=5,x=10,v=5,i=1]
382 ||| val=4
383 |||+ exp=-(z,-(x,y))
384 ||| env=[x=4,y=2,x=3,z=5,x=10,v=5,i=1]
385 |||+ exp=z
386 ||| env=[x=4,y=2,x=3,z=5,x=10,v=5,i=1]
387 ||| val=5
388 |||+ exp=-(x,y)
389 ||| env=[x=4,y=2,x=3,z=5,x=10,v=5,i=1]
390 |||+ exp=x
391 ||| env=[x=4,y=2,x=3,z=5,x=10,v=5,i=1]
392 ||| val=4
393 |||+ exp=y
394 ||| env=[x=4,y=2,x=3,z=5,x=10,v=5,i=1]
395 ||| val=2
396 ||| val=2
397 ||| val=3
398 ||| val=3
399 ||- val=3
400 ||- val=3
401 ||- val=3
402 3
403 >
404 [dshin@acacia let]$ Let -t /home/pl/hw06/tests/let-shadow-body
405 Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.
406 > % =====
407 % shadowing in let. (value=4)
408
409 let x=3
410 in let x=4
411   in x
412
413 % =====
414 |+ exp=let x=3 in let x=4 in x
415 || env=[x=10,v=5,i=1]
416 ||+ exp=3
417 || env=[x=10,v=5,i=1]
418 ||- val=3
419 ||+ exp=let x=4 in x
420 || env=[x=3,x=10,v=5,i=1]
421 |||+ exp=4
422 ||| env=[x=3,x=10,v=5,i=1]
423 ||| val=4
424 |||+ exp=x
425 ||| env=[x=4,x=3,x=10,v=5,i=1]

```

Nov 07, 16 0:17

hw06-LET-output-trace

Page 8/10

```

426 |||- val=4
427 ||- val=4
428 ||- val=4
429 4
430 >
431 [dshin@acacia let]$ Let -t /home/pl/hw06/tests/let-shadow-rhs
432 Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.
433 > % =====
434 % shadowing in let. (value=2)
435
436 let x=3
437 in let x=-(x,1)
438   in x
439
440 % =====
441 |+ exp=let x=3 in let x=-(x,1) in x
442 || env=[x=10,v=5,i=1]
443 ||+ exp=3
444 || env=[x=10,v=5,i=1]
445 ||- val=3
446 ||+ exp=let x=-(x,1) in x
447 || env=[x=3,x=10,v=5,i=1]
448 |||+ exp=-(x,1)
449 ||| env=[x=3,x=10,v=5,i=1]
450 |||+ exp=x
451 ||| env=[x=3,x=10,v=5,i=1]
452 ||| val=3
453 |||+ exp=1
454 ||| env=[x=3,x=10,v=5,i=1]
455 ||| val=1
456 ||| val=2
457 |||+ exp=x
458 ||| env=[x=2,x=3,x=10,v=5,i=1]
459 ||| val=2
460 ||- val=2
461 ||- val=2
462 2
463 >
464 [dshin@acacia let]$ Let -t /home/pl/hw06/tests/let-var
465 Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.
466 > % =====
467 % make sure the let var is evaluated. (value=7)
468
469 let x=-(8,1) in x
470
471 % =====
472 |+ exp=let x=-(8,1) in x
473 || env=[x=10,v=5,i=1]
474 ||+ exp=-(8,1)
475 || env=[x=10,v=5,i=1]
476 |||+ exp=8
477 ||| env=[x=10,v=5,i=1]
478 ||| val=8
479 |||+ exp=1
480 ||| env=[x=10,v=5,i=1]
481 ||| val=1
482 ||- val=7
483 ||+ exp=x
484 || env=[x=7,x=10,v=5,i=1]
485 ||- val=7
486 ||- val=7
487 7
488 >
489 [dshin@acacia let]$ Let -t /home/pl/hw06/tests/variable
490 Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.
491 > % =====

```

Nov 07, 16 0:17	hw06-LET-output-trace	Page 9/10
492	% variable, where env=[i=1,v=5,x=10]. (value=10)	
493		
494	x	
495		
496	% =====	
497	+ exp=x	
498	env=[x=10,v=5,i=1]	
499	- val=10	
500	10	
501	>	
502	[dshin@acacia let]\$ Let -t /home/pl/hw06/tests/variable-error-1	
503	Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.	
504	> % =====	
505	% unbound variable, where env=[i=1,v=5,x=10]. (value=error)	
506		
507	foo	
508		
509	% =====	
510	+ exp=foo	
511	env=[x=10,v=5,i=1]	
512	apply-env: undefined foo	
513	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/env.scm:13:22:	
514	(eopl:error (quote apply-env) "undefined ~a" search-sym)	
515	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:23:4: (	
516	let-values (((val) (value-of-1 exp env))) (trace-run-exit val) val)	
517	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:7:28: (	
518	expval->string (run string))	
519	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:7:15: (	
520	format "~a" (expval->string (run string)))	
521	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:7:6: (d	
522	isplay (format "~a" (expval->string (run string))))	
523	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:5:4: (l	
524	et-values ((string) (readfile file))) (display string) (display (format "~a" ((	
525	....) ....))) (newline))	
526		
527	=== context ===	
528	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:21:2: v	
529	alue-of	
530	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:4:2: ru	
531	nfile	
532		
533	>	
534	[dshin@acacia let]\$ Let -t /home/pl/hw06/tests/variable-error-2	
535	Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.	
536	> % =====	
537	% unbound variable, where env=[i=1,v=5,x=10]. (value=error)	
538		
539	-(x,foo)	
540		
541	% =====	
542	+ exp=-(x,foo)	
543	env=[x=10,v=5,i=1]	
544	+ exp=x	
545	env=[x=10,v=5,i=1]	
546	- val=10	
547	+ exp=foo	
548	env=[x=10,v=5,i=1]	
549	apply-env: undefined foo	
550	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/env.scm:13:22:	
551	(eopl:error (quote apply-env) "undefined ~a" search-sym)	
552	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:23:4: (	
553	let-values (((val) (value-of-1 exp env))) (trace-run-exit val) val)	
554	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:38:33:	
555	(expval->num (value-of exp2 env))	
556	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:37:30:	

Nov 07, 16 0:17	hw06-LET-output-trace	Page 10/10
546	(- (expval->num (value-of exp1 env)) (expval->num (value-of exp2 env)))	
547	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:37:21:	
548	(num-val (- (expval->num (value-of exp1 env)) (expval->num (value-of exp2 env)))	
549	)	
550	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:23:4: (	
551	let-values (((val) (value-of-1 exp env))) (trace-run-exit val) val)	
552	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:7:28: (	
553	expval->string (run string))	
554	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:7:15: (	
555	format "~a" (expval->string (run string)))	
556	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:7:6: (d	
557	isplay (format "~a" (expval->string (run string))))	
558	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:5:4: (l	
559	et-values (((string) (readfile file))) (display string) (display (format "~a" ((	
560	....) ....))) (newline))	
561		
562	=== context ===	
563	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:21:2: v	
564	alue-of	
565	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:28:2: v	
566	alue-of-1	
567	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:21:2: v	
568	alue-of	
569	/home/dshin/2012-ea0011-pl/eopl3-programs/eopl3-interpreters/let/run.scm:4:2: ru	
570	nfile	
571		
572	>	
573	[dshin@acacia let]\$ Let -t /home/pl/hw06/tests/zero-false	
574	Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.	
575	> % =====	
576	% zero? (value=#f)	
577		
578	zero?(37)	
579		
580	% =====	
581	+ exp=zero?(37)	
582	env=[x=10,v=5,i=1]	
583	+ exp=37	
584	env=[x=10,v=5,i=1]	
585	- val=37	
586	- val=#f	
587	#f	
588	>	
589	[dshin@acacia let]\$ Let -t /home/pl/hw06/tests/zero-true	
590	Welcome to MzScheme v372 [3m], Copyright (c) 2004-2007 PLT Scheme Inc.	
591	> % =====	
592	% zero? (value=#t)	
593		
594	zero?(0)	
595		
596	% =====	
597	+ exp=zero?(0)	
598	env=[x=10,v=5,i=1]	
599	+ exp=0	
600	env=[x=10,v=5,i=1]	
601	- val=0	
602	- val=#t	
603	#t	
604	>	
605	[dshin@acacia let]\$	