$x,\ y$	metavariables			
$i,\ j,\ k,\ n$	index variables			
e	::=			expression
	|	x		variable
	|	$\lambda x.e$		lambda abstraction
	|	$e\ x$		function application
	|	(e)	S	parenthesisation, syntactic sugar
	|	$e[e'/x]$	M	substitution syntax, meta-level
	|	$\mathsf{let}\ x_1 = e_1,\ \ldots,\ x_n = e_n\ \mathsf{in}\ e$		let-binding
z	::=			value
	|	$\lambda x.e$		
$\Gamma,\ \Delta,\ \Theta,\ \Xi$	::=			heap
	|	$\{x \mapsto e\}$		(single binding)
	|	$(\Gamma, x \mapsto e)$		$x \notin dom(\Gamma)$
	|	$(\Gamma, x_1 \mapsto e_1,\ \ldots,\ x_n \mapsto e_n)$		$\{x_i\}_1^n \cap dom(\Gamma) = \varnothing$
terminals	::=			
	|	λ		
	|	$\Downarrow$		
	|	$\mapsto$		
	|	let		
	|	in		
formula	::=			
	|	*judgement*		
Jop	::=			
	|	$\Gamma : e \Downarrow \Delta : z$		
judgement	::=			
	|	*Jop*		
user_syntax	::=			
	|	x		
	|	i		
	|	e		
	|	z		
	|	Γ		
	|	*terminals*		

$\boxed{\Gamma : e \Downarrow \Delta : z}$

$$\frac{}{\Gamma : \lambda x.e \Downarrow \Gamma : \lambda x.e}\ \text{LAM}$$

$$\frac{\Gamma : e \Downarrow \Delta : \lambda y.e' \qquad \Delta : e'[x/y] \Downarrow \Theta : z}{\Gamma : e\ x \Downarrow \Theta : z}\ \text{APP}$$

$$\frac{\Gamma : e \Downarrow \Delta : z}{(\Gamma, x \mapsto e) : x \Downarrow (\Delta, x \mapsto z) : z}\ \text{VAR}$$

$$\frac{(\Gamma, x_1 \mapsto e_1, \ldots, x_n \mapsto e_n) : e \Downarrow \Delta : z}{\Gamma : \mathsf{let}\ x_1 = e_1, \ldots, x_n = e_n\ \mathsf{in}\ e \Downarrow \Delta : z}\ \textsc{let}$$

```
Definition rules:         4 good    0 bad
Definition rule clauses: 8 good    0 bad
```