Integers	i		
Identifiers	x, f, C, l		
Unary operators	u	$::= - \mid !$	
Binary operators	b	$::= + \mid - \mid * \mid / \mid \% \mid ==$	
		$\mid\ != \mid < \mid \leq \mid > \mid \geq$	
Programs	p	$::= \overline{d}\, e$	
Declarations	d	$::= \mathrm{def}\ f(\overline{x} : \overline{T}) : T = e;$	Function declaration
		$\mid\ \mathrm{class}\ C\ \{\mathrm{val}\ \overline{l} : \overline{T};\}$	Class declaration
Statements	s	$::= \mathrm{while}\,(e)\,\{\overline{s}\}$	Loop
		$\mid\ \mathrm{var}\, x : T = e;$	Variable declaration
		$\mid\ \mathrm{let}\, x = e;$	Variable assignment
		$\mid\ \mathrm{do}\, e;$	Execution
Expressions	e	$::= i$	Integer
		$\mid\ \mathrm{null}$	Null reference
		$\mid\ x$	Identifier
		$\mid\ \mathrm{if}\,(e_1)\, e_2\, \mathrm{else}\, e_3$	Conditional
		$\mid\ u\, e$	Unary operation
		$\mid\ e_1\, b\, e_2$	Binary operation
		$\mid\ e(\overline{e})$	Function application
		$\mid\ \mathrm{new}\ C(\overline{e})$	Object creation
		$\mid\ e.l$	Field selection
		$\mid\ \{\overline{s}\, e\}$	Block
External types	T	$::= \mathrm{Int}$	Integer type
		$\mid\ (\overline{T}) \Rightarrow T$	Function type
		$\mid\ C$	Object type (for class C)
Internal types	S	$::= T$	External type
		$\mid\ \mathrm{Null}$	Null type
		$\mid\ /\Gamma/$	Class type (scope of fields)
Kinds	k	$::= \mathrm{Var} \mid \mathrm{Fun} \mid \mathrm{Class} \mid \mathrm{Field}$	
Binding	B	$::= x : k[S]$	
Environment	Γ	$::= \overline{B}$	

Notation	Interpretation	Condition
$\overline{x}$	sequence $x_1, \ldots, x_n$	$n \geq 0$
ϵ	empty sequence	
$\lvert\overline{x}\rvert$	length of sequence $\overline{x}$	
$\overline{x} : \overline{T}$	$x_1 : T_1, \ldots, x_n : T_n$	$n = \lvert\overline{x}\rvert = \lvert\overline{T}\rvert \geq 0$
$\Gamma \vdash \overline{e} : \overline{S}$	$\Gamma \vdash e_1 : S_1, \ldots, \Gamma \vdash e_n : S_n$	$n = \lvert\overline{e}\rvert = \lvert\overline{S}\rvert \geq 0$
$\max(S_1, S_2)$	S_1 if $S_2 <: S_1$, S_2 if $S_1 <: S_2$	S_1 and S_2 comparable
$\mathrm{dom}(\Gamma)$	$\{x \mid \exists\, k, S \text{ s.t. } x : k[S] \in \Gamma\}$	
$\Gamma\{x : k[S]\}$	$\Gamma, x : k[S]$	$x \notin \mathrm{dom}(\Gamma)$
$\Gamma\{\epsilon\}$	Γ	
$\Gamma\{E, \overline{E}\}$	$\Gamma\{E\}\{\overline{E}\}$	

Fig. 1. Eins Abstract Syntax and Notations

Expression typing $(\Gamma \vdash e : S)$

$$(\text{IntLit})\frac{}{\Gamma \vdash i : \text{Int}}$$

$$(\text{NullLit})\frac{}{\Gamma \vdash \text{null} : \text{Null}}$$

$$(\text{Ident})\frac{x : k[S] \in \Gamma \qquad k = \text{Var, Fun}}{\Gamma \vdash x : S}$$

$$(\text{If})\frac{\Gamma \vdash e : \text{Int} \\ \Gamma \vdash e_1 : S_1 \qquad \Gamma \vdash e_2 : S_2}{\Gamma \vdash \text{if}\,(e)\, e_1 \,\text{else}\, e_2 : \max(S_1, S_2)}$$

$$(\text{Unop})\frac{\Gamma \vdash e : \text{Int}}{\Gamma \vdash u\, e : \text{Int}}$$

$$(\text{Binop})\frac{\Gamma \vdash e_1 : \text{Int} \qquad \Gamma \vdash e_2 : \text{Int}}{\Gamma \vdash e_1\, b\, e_2 : \text{Int}}$$

$$(\text{RefComp})\frac{b = (!=), (==) \\ \Gamma \vdash e_1 : S_1 \qquad S_1 <: C \\ \Gamma \vdash e_2 : S_2 \qquad S_2 <: C}{\Gamma \vdash e_1\, b\, e_2 : \text{Int}}$$

$$(\text{Apply})\frac{\Gamma \vdash e : (\overline{T}) \Rightarrow T \\ \Gamma \vdash \overline{e} : \overline{S} \\ \overline{S} <: \overline{T}}{\Gamma \vdash e(\overline{e}) : T}$$

$$(\text{Select})\frac{\Gamma \vdash e : C \\ C : \text{Class}[/\Gamma'/] \in \Gamma \\ l : \text{Field}[S] \in \Gamma'}{\Gamma \vdash e.l : S}$$

$$(\text{New})\frac{C : \text{Class}[/\overline{l} : \text{Field}[\overline{S}]/] \in \Gamma \\ \Gamma \vdash \overline{e} : \overline{S'} \\ \overline{S'} <: \overline{S}}{\Gamma \vdash \text{new}\; C(\overline{e}) : C}$$

$$(\text{Block})\frac{n = |\overline{s}| \qquad \Gamma_1 = \Gamma \\ \forall\, i \in [1..n],\ \Gamma_i \vdash s_i \Rightarrow \Gamma_{i+1} \\ \Gamma_{n+1} \vdash e : S}{\Gamma \vdash \{\overline{s}\, e\} : S}$$

Subtyping $(S <: S')$

$$(\text{SRefl})\frac{}{S <: S}$$

$$(\text{SNull})\frac{}{\text{Null} <: C}$$

Statement well-formedness $(\Gamma \vdash s \Rightarrow \Gamma')$

$$(\text{While})\frac{\Gamma \vdash e : \text{Int} \\ n = |\overline{s}| \qquad \Gamma_1 = \Gamma \\ \forall\, i \in [1..n],\ \Gamma_i \vdash s_i \Rightarrow \Gamma_{i+1}}{\Gamma \vdash \text{while}\,(e)\,\{\overline{s}\} \Rightarrow \Gamma}$$

$$(\text{Var})\frac{\Gamma \vdash T \diamond \qquad \Gamma \vdash e : S \qquad S <: T}{\Gamma \vdash \text{var}\, x : T = e; \Rightarrow \Gamma\{x : \text{Var}[T]\}}$$

$$(\text{Let})\frac{x : \text{Var}[S] \in \Gamma \\ \Gamma \vdash e : S' \qquad S' <: S}{\Gamma \vdash \text{let}\, x = e; \Rightarrow \Gamma}$$

$$(\text{Do})\frac{\Gamma \vdash e : T}{\Gamma \vdash \text{do}\, e; \Rightarrow \Gamma}$$

Declaration well-formedness $(\Gamma \vdash d \Rightarrow \Gamma')$

$$(\text{Def})\frac{T' = (\overline{T}) \Rightarrow T \\ \Gamma \vdash T' \diamond \\ \Gamma' = \Gamma\{f : \text{Fun}[T']\} \\ \Gamma'\{\overline{x} : \text{Var}[\overline{T}]\} \vdash e : S \\ S <: T}{\Gamma \vdash \text{def}\, f(\overline{x} : \overline{T}) : T = e; \Rightarrow \Gamma'}$$

$$(\text{Class})\frac{\Gamma\{C : \text{Class}[\epsilon]\} \vdash \overline{T} \diamond \\ \Gamma' = \Gamma\{C : \text{Class}[\epsilon\{\overline{l} : \text{Field}[\overline{T}]\}]\}}{\Gamma \vdash \text{class}\; C\; \{\text{val}\; \overline{l} : \overline{T};\} \Rightarrow \Gamma'}$$

Program well-formedness $(\vdash p \diamond)$

$$(\text{Program})\frac{n = |\overline{d}| \qquad \Gamma_1 = \epsilon \\ \forall\, i \in [1..n],\ \Gamma_i \vdash d_i \Rightarrow \Gamma_{i+1} \\ \Gamma_{n+1} \vdash e : S}{\vdash \overline{d}\, e \diamond}$$

Type well-formedness $(\Gamma \vdash T \diamond)$

$$(\text{IntType})\frac{}{\Gamma \vdash \text{Int} \diamond}$$

$$(\text{FunType})\frac{\Gamma \vdash \overline{T} \diamond \qquad \Gamma \vdash T \diamond}{\Gamma \vdash (\overline{T}) \Rightarrow T \diamond}$$

$$(\text{ObjectType})\frac{C : \text{Class}[S] \in \Gamma}{\Gamma \vdash C \diamond}$$

Fig. 2. Typing rules