

$$\frac{-}{env \triangleright CstString(st) \Rightarrow String(st)}$$

$$\frac{env \triangleright e \Rightarrow Int(n)}{env \triangleright Succ(e) \Rightarrow Int(n + 1)}$$

$$\frac{env \triangleright e \Rightarrow Int(n)}{env \triangleright Prec(e) \Rightarrow Int(n - 1)}$$

$$\frac{env \triangleright e1 \Rightarrow Bool(b1); env \triangleright e2 \Rightarrow Bool(b2)}{env \triangleright And(e1, e2) \Rightarrow Bool(b1 \wedge b2)}$$

$$\frac{env \triangleright e1 \Rightarrow Bool(b1); env \triangleright e2 \Rightarrow Bool(b2)}{env \triangleright Or(e1, e2) \Rightarrow Bool(b1 \vee b2)}$$

$$\frac{-}{env \triangleright Empty(\tau) \Rightarrow Set(\tau, [])}$$

$$\frac{env \triangleright e \Rightarrow (v:\tau)}{env \triangleright Singleton(e) \Rightarrow Set(\tau, [v])}$$

$$\frac{env \triangleright e1 \Rightarrow Set(\tau1, l1); env \triangleright e2 \Rightarrow (v:\tau2) \quad \tau1 = \tau2}{env \triangleright Insert(e1, e2) \Rightarrow Set(\tau1, v \times l1)}$$

$$\frac{env \triangleright e1 \Rightarrow Set(TVoid, l1); env \triangleright e2 \Rightarrow (v:\tau2) \quad isempty(l1)}{env \triangleright Insert(e1, e2) \Rightarrow Set(\tau2, [v])}$$

$$\frac{env \triangleright e1 \Rightarrow Set(\tau1, l1); env \triangleright e2 \Rightarrow (v:\tau2) \quad \tau1 = \tau2}{env \triangleright Remove(e1, e2) \Rightarrow Set(\tau, v \times l1)}$$

$$\frac{length(l) = 1 \quad env \triangleright Singleton(first(l)) \Rightarrow Set(\tau, l)}{env \triangleright Of(l) \Rightarrow Set(\tau, l)}$$

$$\frac{\text{length}(l) > 1 \quad env \triangleright \text{Insert}(\text{Of}(\text{removefirst}(l)), \text{first}(l)) \Rightarrow \text{Set}(\tau, l)}{env \triangleright \text{Of}(l) \Rightarrow \text{Set}(\tau, l)}$$

$$\frac{env \triangleright e1 \Rightarrow \text{Set}(\tau1, l1); env \triangleright e2 \Rightarrow \text{Set}(\tau2, l2); \tau1 = \tau2 \quad (\forall x \in l3: x \in l1 \vee x \in l2); (\forall x \in l1 \vee x \in l2 : x \in l3) \quad (\forall x \in l3 : \text{car}(\{y \in l3 : y = x\}) = 1)}{env \triangleright \text{UnionSet}(e1, e2) \Rightarrow \text{Set}(\tau, l3)}$$

$$\frac{env \triangleright e1 \Rightarrow \text{Set}(\tau1, l1); env \triangleright e2 \Rightarrow \text{Set}(\tau2, l2); \tau1 = \tau2 \quad (\forall x \in l3: x \in l1 \wedge x \in l2); (\forall x \in l1 \wedge x \in l2 : x \in l3) \quad (\forall x \in l3 : \text{car}(\{y \in l3 : y = x\}) = 1)}{env \triangleright \text{InterSet}(e1, e2) \Rightarrow \text{Set}(\tau, l3)}$$

$$\frac{env \triangleright e1 \Rightarrow \text{Set}(\tau1, l1); env \triangleright e2 \Rightarrow \text{Set}(\tau2, l2); \tau1 = \tau2 \quad (\forall x \in l3: x \in l1 \wedge x \notin l2); (\forall x \in l1 \wedge x \notin l2 : x \in l3) \quad (\forall x \in l3 : \text{car}(\{y \in l3 : y = x\}) = 1)}{env \triangleright \text{DiffSet}(e1, e2) \Rightarrow \text{Set}(\tau, l3)}$$

$$\frac{env \triangleright e1 \Rightarrow \text{Set}(\tau, l)}{env \triangleright \text{IsEmpty}(e1) \Rightarrow \text{Bool}(\text{isempty}(l))}$$

$$\frac{env \triangleright e1 \Rightarrow \text{Set}(\tau1, l); env \triangleright e2 \Rightarrow (v: \tau2); \tau1 = \tau2}{env \triangleright \text{Contains}(e1, e2) \Rightarrow \text{Bool}(\text{contains}(l, v))}$$

$$\frac{env \triangleright e1 \Rightarrow \text{Set}(\tau1, l); env \triangleright e2 \Rightarrow (v: \tau2); \tau1 = \tau2}{env \triangleright \text{SubSet}(e1, e2) \Rightarrow \text{Bool}(\text{issubset}(l1, l2))}$$

$$\frac{env \triangleright e1 \Rightarrow \text{Set}(\tau, l); \neg \text{empty}(l)}{env \triangleright \text{Max}(e1) \Rightarrow (\text{max}(l))}$$

$$\frac{env \triangleright e1 \Rightarrow \text{Set}(\tau, l); \neg \text{empty}(l)}{env \triangleright \text{Min}(e1) \Rightarrow (\text{min}(l))}$$

$$\frac{env \triangleright e1 \Rightarrow \text{Set}(\tau, l); \neg \text{empty}(l)}{env \triangleright \text{Decapitate}(e1) \Rightarrow \text{Set}(\tau, \text{first}(l) \times l)}$$

$$\frac{env \triangleright e1 \Rightarrow Set(\tau, l); empty(l)}{env \triangleright Decapitate(e1) \Rightarrow Set(\tau, l)}$$

$$\frac{env \triangleright e2 \Rightarrow Set(\tau, l); env \triangleright e1 \Rightarrow Closure("x", b, envF) \quad envF \left[first(l)/_x \right] \triangleright b \Rightarrow v}{env \triangleright ApplyFirst(e1, e2) \Rightarrow v}$$

$$\frac{env \triangleright e2 \Rightarrow Set(\tau, l); env \triangleright e1 \Rightarrow RecClosure("f", "x", b, envF) \quad envF \left[first(l)/_x; RecClosure("f", "x", b, envF)/_f \right] \triangleright b \Rightarrow v}{env \triangleright ApplyFirst(e1, e2) \Rightarrow v}$$

$$\frac{env \triangleright e2 \Rightarrow Set(\tau, l); \neg empty(l) \quad env \triangleright And(ApplyFirst(e1, e2), ForAll(e1, Decapitate(e2))) \Rightarrow Bool(b)}{env \triangleright ForAll(e1, e2) \Rightarrow Bool(b)}$$

$$\frac{env \triangleright e2 \Rightarrow Set(\tau, l); empty(l)}{env \triangleright ForAll(e1, e2) \Rightarrow Bool(true)}$$

$$\frac{env \triangleright e2 \Rightarrow Set(\tau, l); \neg empty(l) \quad env \triangleright Or(ApplyFirst(e1, e2), Exist(e1, Decapitate(e2))) \Rightarrow Bool(b)}{env \triangleright Exist(e1, e2) \Rightarrow Bool(b)}$$

$$\frac{env \triangleright e2 \Rightarrow Set(\tau, l); empty(l)}{env \triangleright Exist(e1, e2) \Rightarrow Bool(false)}$$

$$\frac{env \triangleright e2 \Rightarrow Set(\tau_1, l_1); length(l) > 0 \quad env \triangleright Insert(Map(e1, Decapitate(e2)), ApplyFirst(e1, e2)) \Rightarrow Set(\tau_2, l_2)}{env \triangleright Map(e1, e2) \Rightarrow Set(\tau_2, l_2)}$$

$$\frac{env \triangleright e2 \Rightarrow Set(\tau, l); length(l) = 0}{env \triangleright Map(e1, e2) \Rightarrow Set(TVoid, [])}$$

$$\frac{env \triangleright ApplyFirst(e1, e2) \Rightarrow Bool(true) \quad env \triangleright Insert(Filter(e1, Decapitate(e2)), Min(e2)) \Rightarrow Set(\tau, l_1)}{env \triangleright Filter(e1, e2) \Rightarrow Set(\tau, l_1)}$$

$$\begin{array}{c}
env \triangleright ApplyFirst(e1, e2) \Rightarrow Bool(false) \\
env \triangleright Filter(e1, Decapitate(e2)) \Rightarrow Set(\tau, l1) \\
\hline
env \triangleright Filter(e1, e2) \Rightarrow Set(\tau, l1) \\
\\
env \triangleright InEmpty(e2) \Rightarrow Bool(true) \\
env \triangleright e2 \Rightarrow Set(\tau, l) \\
\hline
env \triangleright Filter(e1, e2) \Rightarrow Set(\tau, l)
\end{array}$$

Note:

- $a \times b$ = la lista ordinata generata dall'aggiunta di a nella lista ordinata b;
- $a \div b$ = la lista ordinata generata dalla rimozione di a dalla lista ordinata b;
- $first(l)$ = primo elemento della lista l;
- $length(l)$ = lunghezza della lista l;
- $removefirst(l)$ = restituisce la lista l senza il primo elemento
- $car(i)$ = cardinalità dell'insieme i;
- $isempty(l)$ = restituisce true se la lista ordinata l è vuota, false altrimenti;
- $contains(l, e)$ = restituisce true se la lista ordinata l contiene l'elemento e, false altrimenti;
- $issubset(l1, l2)$ = restituisce true se la lista ordinata l1 è contenuta interamente nella lista ordinata l2, false altrimenti;
- \neg = not logico;
- $min(l)$ = restituisce L'elemento minimo contenuto nella lista l
- $max(l)$ = restituisce L'elemento massimo contenuto nella lista l