Regole operazionali

- t1 ∈ typeSet → per dire che “t1” appartiene all’insieme dei tipi consentiti per i Set (Insiemi)
- sameType(a,b) → per dire che “a” e “b” sono dello stesso tipo
- card(v,l1) → è una funzione che restituisce la cardinalità dell’elemento “v” all’interno di “l1”
- “=” → restituisce “true” sse i due valori sono uguali, “false” altrimenti
- “∪”, “∩”, “-” → operatori insiemistici di unione, intersezione e differenza

Definizione del tipo Set:

Set(list,type):

$$\frac{\mathrm{t1} \in \mathrm{typeSet}, \mathrm{Env} \rhd \mathrm{e} \rightarrow \mathrm{l1}, (\forall\, \mathrm{x} \in \mathrm{l1}\,.\ \mathrm{sameType(x,t1)} \wedge \mathrm{card(x,l1)} = 1 \vee \mathrm{card(x,l1)} = 0)}{\mathrm{Env} \rhd \mathrm{Set(l1,t1)}}$$

Definizione classi di operazione:

EmptySet(t1):

$$\frac{\mathrm{t1} \in \mathrm{typeSet}}{\mathrm{Env} \rhd \mathrm{EmptySet(i)} \rightarrow \mathrm{Set([\,],t1)}}$$

Singleton(v1):

$$\frac{\mathrm{t1} \in \mathrm{typeSet}, \mathrm{Env} \rhd \mathrm{e} \rightarrow \mathrm{v1}, \mathrm{sameType(t1,v1)}}{\mathrm{Env} \rhd \mathrm{Singleton(e)} \rightarrow \mathrm{Set([v1],t1)}}$$

Of(t1,collection):

$$\frac{\mathrm{t1} \in \mathrm{typeSet}, \mathrm{e} \in \mathrm{collection}, \mathrm{Env} \rhd \mathrm{e} \rightarrow \mathrm{l1}, (\forall\, \mathrm{x} \in \mathrm{l1}\,.\ \mathrm{sameType(x,t1)} \wedge \mathrm{card(x,l1)} = 1)}{\mathrm{Env} \rhd \mathrm{Of(t1,e)} \rightarrow \mathrm{Set(l1,t1)}}$$

Union(s1,s2):

$$\frac{\mathrm{Env} \rhd \mathrm{e1} \rightarrow \mathrm{Set(l1,id)}, \mathrm{Env} \rhd \mathrm{e2} \rightarrow \mathrm{Set(l2,id)}}{\mathrm{Env} \rhd \mathrm{Union(e1,e2)} \rightarrow \mathrm{Set(l1} \cup \mathrm{l2,id)}}$$

Intersection(s1,s2):

$$\frac{\mathrm{Env} \rhd \mathrm{e1} \rightarrow \mathrm{Set(l1,id)}, \mathrm{Env} \rhd \mathrm{e2} \rightarrow \mathrm{Set(l2,id)}}{\mathrm{Env} \rhd \mathrm{Intersection(e1,e2)} \rightarrow \mathrm{Set(l1} \cap \mathrm{l2,id)}}$$

Difference(s1,s2):

$$\frac{\mathrm{Env} \rhd \mathrm{e1} \rightarrow \mathrm{Set(l1,id)}, \mathrm{Env} \rhd \mathrm{e2} \rightarrow \mathrm{Set(l2,id)}}{\mathrm{Env} \rhd \mathrm{Difference(e1,e2)} \rightarrow \mathrm{Set(\,l1} - \mathrm{l2,id)}}$$

Push(s1,v):

$$\frac{\mathrm{Env} \rhd \mathrm{e1} \rightarrow \mathrm{Set(l1, id)}, \mathrm{Env} \rhd \mathrm{e2} \rightarrow \mathrm{v1}, \mathrm{sameType(id, v1)}}{\mathrm{Env} \rhd \mathrm{Push(e1, e2)} \rightarrow \mathrm{Set(lst} \cup \mathrm{v1, id)}}$$

RemoveFrom(s1,v):

$$\frac{\mathrm{Env} \rhd \mathrm{e1} \rightarrow \mathrm{Set(l1, id)}, \mathrm{Env} \rhd \mathrm{e2} \rightarrow \mathrm{v1}, \mathrm{sameType(id, v1)}}{\mathrm{Env} \rhd \mathrm{RemoveFrom(e1, e2)} \rightarrow \mathrm{Set(lst} - \mathrm{v1, id)}}$$

IsEmpty(s1):

$$\frac{\mathrm{Env} \rhd \mathrm{s1} \rightarrow \mathrm{Set(l1, id)}}{\mathrm{Env} \rhd \mathrm{IsEmpty(s1)} \rightarrow \mathrm{s1} = \mathrm{Set([\quad], id)}}$$

Contains(s1,v):

$$\frac{\mathrm{Env} \rhd \mathrm{e1} \rightarrow \mathrm{Set(l1, id)}, \mathrm{Env} \rhd \mathrm{e2} \rightarrow \mathrm{v1}, \mathrm{sameType(id, v1)}}{\mathrm{Env} \rhd \mathrm{Contains(e1, e2)} \rightarrow ((\mathrm{l1} \cap \mathrm{v1}) = \mathrm{v1})}$$

Subset(s1,s2):

$$\frac{\mathrm{Env} \rhd \mathrm{e1} \rightarrow \mathrm{Set(l1, id)}, \mathrm{Env} \rhd \mathrm{e2} \rightarrow \mathrm{Set(l2, id)}}{\mathrm{Env} \rhd \mathrm{Subset(e1, e2)} \rightarrow ((\mathrm{l1} \cap \mathrm{l2}) = \mathrm{l2})}$$

Max(s1):

$$\frac{\mathrm{Env} \rhd \mathrm{e} \rightarrow \mathrm{Set(l1, id)}, (\exists\, x \in \mathrm{l1} .\ (\forall\, y \in \mathrm{l1} . x > y))}{\mathrm{Env} \rhd \mathrm{Max(e)} \rightarrow x}$$

Min(s1):

$$\frac{\mathrm{Env} \rhd \mathrm{e} \rightarrow \mathrm{Set(l1, id)}, (\exists\, x \in \mathrm{l1} .\ (\forall\, y \in \mathrm{l1} . x < y))}{\mathrm{Env} \rhd \mathrm{Min(e)} \rightarrow x}$$

For_all(pred,set):

$$\frac{\begin{array}{c}\mathrm{Env} \rhd \mathrm{e1} \rightarrow \mathrm{Closure(arg, fbody, fDecEnv)}, \mathrm{Env} \rhd \mathrm{e2} \rightarrow \mathrm{Set(l1, id)} \\ \big(\forall x \in \mathrm{l1} . \mathrm{fDecEnv[l \backslash x]} \rhd \mathrm{fbody} \rightarrow \mathrm{sameType(v, Bool)}\big), \big(\forall\, y . y = \mathrm{Bool(true)}\big) \rightarrow \mathrm{ret}\end{array}}{\mathrm{Env} \rhd \mathrm{Exists(e1, e2)} \rightarrow \mathrm{ret}}$$

Exist(pred,set):

$$\frac{\begin{array}{c}\mathrm{Env} \rhd \mathrm{e1} \rightarrow \mathrm{Closure(arg, fbody, fDecEnv)}, \mathrm{Env} \rhd \mathrm{e2} \rightarrow \mathrm{Set(l1, id)} \\ \big(\forall x \in \mathrm{l1} . \mathrm{fDecEnv[l \backslash x]} \rhd \mathrm{fbody} \rightarrow \mathrm{sameType(v, Bool)}\big), \big(\exists\, y . y = \mathrm{Bool(true)}\big) \rightarrow \mathrm{ret}\end{array}}{\mathrm{Env} \rhd \mathrm{Exists(e1, e2)} \rightarrow \mathrm{ret}}$$

Filter(pred,set):

$$\frac{\begin{array}{c}\mathrm{Env}\ \rhd\ \mathrm{e1}\ \rightarrow\ \mathrm{Closure(arg, fbody, fDecEnv), Env}\ \rhd\ \mathrm{e2}\ \rightarrow\ \mathrm{Set(l1, id)} \\ \left(\forall \mathrm{x} \in \mathrm{l1} \,.\, \mathrm{fDecEnv[l\backslash x]}\ \rhd\ \mathrm{fbody}\ \rightarrow \mathrm{sameType(v, Bool)}\right), (\forall\, \mathrm{y}\ \in\ \mathrm{l1}. (\mathrm{z}\ =\ \mathrm{Bool(true)})\ \rightarrow\ \mathrm{y}) \rightarrow\ \mathrm{ret}\end{array}}{\mathrm{Env}\ \rhd\ \mathrm{Filter(e1, e2)} \rightarrow\ \mathrm{ret}}$$

Map(funct,set):

$$\frac{\begin{array}{c}\mathrm{Env}\ \rhd\ \mathrm{e1}\ \rightarrow\ \mathrm{Closure(arg, fbody, fDecEnv), Env}\ \rhd\ \mathrm{e2}\ \rightarrow\ \mathrm{Set(l1, id)} \\ (\forall \mathrm{x} \in \mathrm{l1} \,.\, \mathrm{fDecEnv[l\backslash x]}\ \rhd \mathrm{fbody} \rightarrow\ \mathrm{y})\ \rightarrow\ \mathrm{ret}\end{array}}{\mathrm{Env}\ \rhd\ \mathrm{Map(e1, e2)} \rightarrow\ \mathrm{ret}}$$