Relazione Progetto OCAML

Niccolò Campitelli – 583283

SCELTE IMPLEMENTATIVE

E' stato introdotto il tipo di elementi omogenei accettati dal singolo insieme:
$type\ \boldsymbol{set\_t}\ =\ SET\_INT\ |\ SET\_BOOL\ |\ SET\_STRING$

Sono stati introdotti a Run-Time i tipi:
$type\ evT\ =\ |\ \boldsymbol{String}\ \ of\ \ string\quad |\ \boldsymbol{Set}\ \ of\ \ (evT\ list) * set\_t$

Il tipo Set è quindi implementato a tempo di esecuzione come una coppia di elementi:

- Una lista di elementi evT che rappresenta astrattamente un insieme senza duplicati;
- Un'istanza di set_t che stabilisce l'unico tipo accettato nella lista di elementi.

La sintassi del linguaggio è stata estesa introducendo i due costruttori per il tipo Set:
$type\ exp\ =\ |\ \boldsymbol{Empty}\ \ of\ \ set\_t\quad |\ \boldsymbol{Singleton}\ \ of\ \ exp * set\_t$

E il costruttore per il tipo $String$:
$type\ exp\ =\ |\ \boldsymbol{CstString}\ \ of\ \ string$

Oltre alle nuove operazioni richieste per il tipo Set, sono state introdotte o estese nella sintassi:

- la divisione ($\boldsymbol{Div}$) e il modulo ($\boldsymbol{Mod}$) tra coppie di interi;
- la concatenazione ($\boldsymbol{Concat}$) tra coppie di stringhe;
- il maggiore ($\boldsymbol{Major}$) tra coppie di interi, booleani e stringhe;
- l'uguaglianza ($\boldsymbol{Eq}$) tra coppie di interi, booleani, stringhe e insiemi.

Per un'implementazione più snella delle operazioni sui Set è stata utilizzata la libreria $List.mem$.
Per una maggiore comprensione personale sono state rinominate alcune espressioni e tipi dato.

Le regole operazionali presentate nelle pagine seguenti astraggono dalla lista di elementi in Set trattandola come un insieme. Inoltre sono esplicitati i tipi a Run-Time e gli esiti del typechecker.

La batteria di test è presente in fondo al file dell'interprete.

TYPECHECKER STATICO

Il typechecker presente nella versione base dell'interprete è stato esteso per la verifica dei tipi $\boldsymbol{String, Set, Closure}$ ed è invocato rispettivamente con le stringhe "$string$", "set", "fun".

E' stata inoltre introdotta la funzione $\boldsymbol{set\_typecheck}$ di supporto al typechecker che prende come argomenti un set_t e un evT. Restituisce $true\ SSE$ l'elemento di tipo evT è ospitabile nell'insieme di tipo set_t (omogeneità rispettata).

REGOLE OPERAZIONALI - TIPO SET

$$\frac{(t : set\_t)}{(env \triangleright \textbf{Empty}(\text{t})) \Rightarrow \text{Set}(\emptyset,\ \text{t})} \qquad \frac{(env \triangleright \text{e} \Rightarrow \text{v}),\ (t : set\_t),\ (set\_typecheck(t, v) = \text{true})}{(env \triangleright \textbf{Singleton}(e, t)) \Rightarrow \text{Set}(\{\text{v}\},\ \text{t})}$$

REGOLE OPERAZIONALI – OPERAZIONI SU SET

$$\frac{(env \triangleright \text{e} \Rightarrow \text{s}),\ (typecheck(\text{"}set\text{"}, s) = \text{true}),\ (s = Set(l, t))}{(env \triangleright \textbf{IsEmpty}(e)) \Rightarrow \text{Bool}(\text{l} = \emptyset)}$$

$$\frac{(env \triangleright \text{e1} \Rightarrow \text{s}),\ (env \triangleright \text{e2} \Rightarrow \text{x}),\ (typecheck(\text{"set"}, s) = \text{true}),\ (s = Set(l, t)),\ (set\_typecheck(t, x) = \text{true})}{(env \triangleright \textbf{Contains}(e1, e2)) \Rightarrow \text{Bool}(\text{x} \in \text{l})}$$

$$\frac{(env \triangleright \text{e1} \Rightarrow \text{s}),\ (env \triangleright \text{e2} \Rightarrow \text{x}),\ (typecheck(\text{"set"}, s) = \text{true}),\ (s = Set(l, t)),\ (set\_typecheck(t, x) = \text{true})}{(env \triangleright \textbf{Push}(e1, e2)) \Rightarrow \text{Set}((\text{l} \cup \{\text{x}\}),\ \text{t})}$$

$$\frac{(env \triangleright \text{e1} \Rightarrow \text{s}),\ (env \triangleright \text{e2} \Rightarrow \text{x}),\ (typecheck(\text{"set"}, s) = \text{true}),\ (s = Set(l, t)),\ (set\_typecheck(t, x) = \text{true})}{(env \triangleright \textbf{Pop}(e1, e2)) \Rightarrow \text{Set}((\text{l} \setminus \{\text{x}\}),\ \text{t})}$$

$$\frac{(env \triangleright \text{e} \Rightarrow \text{s}),\ (typecheck(\text{"set"}, s) = \text{true}),\ (s = Set(l, t)),\ (l \neq \emptyset)}{(env \triangleright \textbf{Min}(e)) \Rightarrow \min\{\ \text{l}\ \}}$$

$$\frac{(env \triangleright \text{e} \Rightarrow \text{s}),\ (typecheck(\text{"set"}, s) = \text{true}),\ (s = Set(l, t)),\ (l \neq \emptyset)}{(env \triangleright \textbf{Max}(e)) \Rightarrow \max\{\ \text{l}\ \}}$$

$$\frac{(env \triangleright \text{e1} \Rightarrow \text{s1}),\ (env \triangleright \text{e2} \Rightarrow \text{s2}),\ (typecheck(\text{"set"}, s1) = \text{true}),\ (typecheck(\text{"set"}, s2) = \text{true})\ (s1 = Set(l1, t1)),\ (s2 = Set(l2, t2)),\ (t1 = \text{t2})}{(env \triangleright \textbf{IsSubset}(e1, e2)) \Rightarrow \text{Bool}(\text{l1} \subseteq \text{l2})}$$

$$\frac{(env \triangleright \text{e1} \Rightarrow \text{s1}),\ (env \triangleright \text{e2} \Rightarrow \text{s2}),\ (typecheck(\text{"set"}, s1) = \text{true}),\ (typecheck(\text{"set"}, s2) = \text{true}),\ (s1 = Set(l1, t1)),\ (s2 = Set(l2, t2)),\ (t1 = \text{t2})}{(env \triangleright \textbf{Union}(e1, e2)) \Rightarrow \text{Set}((\text{l1} \cup \text{l2}), \text{t1})}$$

$$\frac{(env \triangleright \text{e1} \Rightarrow \text{s1}),\ (env \triangleright \text{e2} \Rightarrow \text{s2}),\ (typecheck(\text{"set"}, s1) = \text{true}),\ (typecheck(\text{"set"}, s2) = \text{true}),\ (s1 = Set(l1, t1)),\ (s2 = Set(l2, t2)),\ (t1 = \text{t2})}{(env \triangleright \textbf{Inter}(e1, e2)) \Rightarrow \text{Set}((\text{l1} \cap \text{l2}), \text{t1})}$$

$$\frac{(env \triangleright \text{e1} \Rightarrow \text{s1}),\ (env \triangleright \text{e2} \Rightarrow \text{s2}),\ (typecheck(\text{"set"}, s1) = \text{true}),\ (typecheck(\text{"set"}, s2) = \text{true}),\ (s1 = Set(l1, t1)),\ (s2 = Set(l2, t2)),\ (t1 = \text{t2})}{(env \triangleright \textbf{Subtract}(e1, e2)) \Rightarrow \text{Set}((\text{l1} \setminus \text{l2}), \text{t1})}$$

$$\frac{(env \triangleright \text{e1} \Rightarrow \text{s}),\ (env \triangleright \text{e2} \Rightarrow \text{p}),\ (typecheck(\text{"set"}, s) = \text{true}),\ (typecheck(\text{"fun"}, p) = \text{true}),\ \big(s = Set(l, t)\big),\ \big(p = Closure(par, body, env')\big),\ (\forall x \in l \Rightarrow (env'[x/par] \triangleright \text{corpo} \Rightarrow \text{ris}) \wedge (typecheck(\text{"bool"}, ris) = true))}{(env \triangleright \textbf{ForAll}(e1, e2)) \Rightarrow \text{Bool}(\forall x \Rightarrow (ris = Bool(true)))}$$

$$\frac{(env \triangleright \text{e1} \Rightarrow \text{s}),\ (env \triangleright \text{e2} \Rightarrow \text{p}),\ (typecheck(\text{"set"}, s) = \text{true}),\ (typecheck(\text{"fun"}, p) = \text{true}),\ \big(s = Set(l, t)\big),\ \big(p = Closure(par, body, env')\big),\ (\forall x \in l \Rightarrow (env'[x/par] \triangleright \text{corpo} \Rightarrow \text{ris}) \wedge (typecheck(\text{"bool"}, ris) = true))}{(env \triangleright \textbf{Exist}(e1, e2)) \Rightarrow \text{Bool}(\exists x\,.\,(\text{ris} = Bool(true)))}$$

La condizione del quantificatore universale è in realtà limitata ai primi $(\text{k} - 1)$ valori controllati dell'insieme, con $k \mid \text{ris} = Bool(true)$ primo valore incontrato che rende vera l'$Exist$.

$$\frac{(env \triangleright \text{e1} \Rightarrow \text{s}),\ (env \triangleright \text{e2} \Rightarrow \text{p}),\ (typecheck(\text{"set"}, s) = \text{true}),\ (typecheck(\text{"fun"}, p) = \text{true}),\ \big(s = Set(l, t)\big),\ \big(p = Closure(par, body, env')\big),\ (\forall x \in l \Rightarrow (env'[x/par] \triangleright \text{corpo} \Rightarrow \text{ris}) \wedge (typecheck(\text{"bool"}, ris) = true))}{(env \triangleright \textbf{Filter}(e1, e2)) \Rightarrow \text{Set}(\{x \mid \text{ris} = Bool(true)\}, \text{t})}$$

$$\frac{(env \triangleright \text{e1} \Rightarrow \text{s}),\ (env \triangleright \text{e2} \Rightarrow \text{p}),\ (typecheck(\text{"set"}, s) = \text{true}),\ (typecheck(\text{"fun"}, p) = \text{true}),\ \big(s = Set(l, t)\big),\ \big(p = Closure(par, body, env')\big),\ (\forall x \in l \Rightarrow (env'[x/par] \triangleright \text{corpo} \Rightarrow \text{ris}) \wedge (set\_typecheck(\text{t}, ris) = true))}{(env \triangleright \textbf{Map}(e1, e2)) \Rightarrow \text{Set}(\{ris\}, \text{t})}$$

Map restituisce solo insiemi con lo stesso tipo di quello preso come argomento.

$$\frac{(env \triangleright \text{e1} \Rightarrow \text{s1}),\ (env \triangleright \text{e2} \Rightarrow \text{s2}),\ (typecheck(\text{"set"}, s1) = \text{true}),\ (typecheck(\text{"set"}, s2) = \text{true}),\ \big(s1 = Set(l1, t1)\big),\ \big(s2 = Set(l2, t2)\big),\ (t1 = t2)}{(env \triangleright \textbf{Eq}(e1, e2)) \Rightarrow \text{Bool}(l1 = l2)}$$

Regola operazionale della Eq circoscritta al caso dei Set (opera anche con $Int/Bool/String$).