A. *Haskell Syntax*

Declaration	d	::=	v :: τ = e
Syntax Node	h	::=	e | τ
Term Variable	v	::=	x, y, z
Type variable	α	::=	a, b, c
Type constant	t	::=	Int | Bool
Expression	e	::=	v | λx.e | e1 e2 | let d in e2
Type	τ	::=	α | t | τ1 → τ2

B. *Prolog Syntax*

Term	**T**	::=	**V | A | C**
Variable	**V**	::=	**X, Y, Z**
Atom	**A**	::=	**p, q, r**
Compound Term	**C**	::=	**A(P1, P2, ...)**
List	**L**	::=	**[]|[T1, T2, ...]|[T|L]**
Clause	**P**	::=	**C ← P1, P2 ...**

C. *Auxiliary Functions*

gen(h, **T**) → **T**

gen_decl(d) → **P**

fresh() → **V**

var(v | α) → **V**

atom(t) → **A**

D. *Constraints Generation Rules*

$$\frac{gen(x,\ \mathbf{V}),\ x \in \Gamma,\ var(x) \Rightarrow \mathbf{V_x}}{\mathbf{unify(V,\ V_x)}}\ [\text{VAR1}]$$

$$\frac{gen(x,\ \mathbf{V}),\ x \notin \Gamma,\ \ x_1,\ x_2, \ldots \in \Gamma,\ var(x_1) \Rightarrow \mathbf{V_1},\ var(x_2) \Rightarrow \mathbf{V_2}, \ldots}{\mathbf{x(V,\ [V_1,\ V_2,\ \ldots|\_])}}\ [\text{VAR2}]$$

$$\frac{gen(\lambda x.e,\ \mathbf{V}),\ var(x) \Rightarrow \mathbf{V_x},\ fresh \Rightarrow \mathbf{V_e}}{\mathbf{unify(V,\ fun(V_x,\ V_e))},\ \ \Gamma,\ x \vdash gen(e,\ \mathbf{V_e})}\ [\text{ABS}]$$

$$\frac{gen(e_1\ e_2,\ \mathbf{V}),\ fresh \Rightarrow \mathbf{V_1},\ fresh \Rightarrow \mathbf{V_2}}{\mathbf{unify(fun(V_2,\ V),\ V_1)},\ \ \Gamma \vdash gen(e_1,\ \mathbf{V_1}),\ gen(e_2,\ \mathbf{V_2})}\ [\text{APP}]$$

$$\frac{gen(t,\ \mathbf{V}),\ atom(t) \Rightarrow \mathbf{A}}{\mathbf{unify(A,\ V)}}\ [\text{TYPE-CON}] \qquad \frac{gen(\alpha,\ \mathbf{V}),\ var(\alpha) \Rightarrow \mathbf{V_\alpha}}{\mathbf{unify(V,\ V_\alpha)}}\ [\text{TYPE-VAR}]$$

$$\frac{gen(\tau_1 \rightarrow \tau_2,\ \mathbf{V})}{\mathbf{unify(V,\ fun(V_1,\ V_2)},\ \ \Gamma \vdash gen(\tau_1,\ \mathbf{V_1}),\ gen(\tau_2,\ \mathbf{V_2})}\ [\text{TYPE-FUN}]$$

$$\frac{gen(\text{let x ::}\tau\text{ = e1 in e2},\ \mathbf{V}),\ fresh \Rightarrow \mathbf{V_1},\ fresh \Rightarrow \mathbf{V_2},\ \Gamma \vdash gen\_decl(x :: \tau = e_1)}{\mathbf{unify(T,\ V_2)},\ \Gamma \vdash gen(e_2\ ,\ \mathbf{V_2})}\ [\text{LET}]$$

E. *Predicates Generation Rules*

$$\frac{gen\_decl(x :: \tau = e),\ x_1, x_2, \ldots \in \Gamma,\ var(x_1) \Rightarrow \mathbf{V_1},\ var(x_2) \Rightarrow \mathbf{V_2}, \ldots}{\mathbf{x(V,\ [V_1,\ V_2, \ldots|\_])} \leftarrow\ \ \Gamma \vdash gen(e,\ \mathbf{V}),\ gen(\tau,\ \mathbf{V}).}\ [\text{DECL}]$$