Type Checking (2)

- Tipi derivati: regole di inferenza
- Coercion e overload: regole di inferenza
- Estendiamo gli attributi della grammatica prec.

Cosa succede quando estendiamo il sistema dei tipi con tipi derivati

regole di inferenza:

$$\frac{e{:}array(i,t) \quad e'{:}i' \quad i \approx i'}{e[e']{:}t}$$

$$\frac{e{:}record(i_1{:}t_1 \ldots i_k{:}t_k) \quad (i=i_j,\ 1 \leq j \leq k)}{e.i{:}t_j}$$

Equivalenza ≈ tra tipi

strutturale:

$$\frac{t_1 \approx t_1' \quad t_k \approx t_k'}{y[t_1,...,t_k] \approx y[t_1',...,t_k']}$$

referenziale

$$\frac{t \equiv t'}{t \approx t'}$$

Coercion e overloading

regole di inferenza:

$$\frac{e:t \quad \text{into}: t \rightarrow t'}{\text{into } e: t'}$$

$$\frac{f:\{t_1 \rightarrow t_1', \ldots, t_k \rightarrow t_k'\} \quad (e_j:t_j \ , 1 \leq j \leq k)}{f(e_j):t_j}$$

Aggiungiamo a Semplice funzioni overloaded:

1) produzioni attributate per le dichiarazioni

2) produzioni attributate per le espressioni

(attenzione alla nuova espressione di tipo {__})

nota: N = neutro rispetto $\times$, quindi $N \times t1 \times \ldots \times tn \times N = t1 \times \ldots \times tn$

[0]**P**::= **Ds Dff Cs**	**P**.r:= **Cs** .r
[1]**P**::= **Cs**	**P**.r:= **Cs** .r
[2]**Ds**::= **Var Dts**	
[3]**Dts**::=**Dt : Y Dts'**	addtype-set(**Dt**.entry,**Y**.t)
[4]**Dts'**$_1$ ::= **; Dt : Y Dts'**$_2$	addtype-set(**Dt**.entry,**Y**.t)
[5] **Dts'** ::= ε	
[6]**Dt**::=**ide O**	**Dt**.entry:=cons(**ide**.entry,**O**.entry)
[7]**O**$_1$::= **, ide O**$_2$	**O**$_1$.entry:=cons(**ide**.entry,**O**$_2$.entry)
[8]**O**::= ε	**O**.entry:=emptylist
[28]**Dff**::= **Fun Df Dff'**	
[29]**Dff'**$_1$::= **; Df Dff'**$_2$	
[30]**Dff'**::= ε	
[31]**Df**::= **funct ide FPP : Y ; P**	overload(**ide**.entry,**FPP**.t→**Y**.t)
[32]**FPP**::= **(FP : Y FPP'**	**FPP**.t:=**Y**.t × **FPP'**.t
[33]**FPP**::= ε	**FPP**.t:=**N**
[34]**FPP'**$_1$::= **, FP : Y FPP'**$_2$	**FPP'**$_1$.t:=**Y**.t × **FPP'**$_2$.t
[35]**FPP'**::= **)**	**FPP'**.t:=**N**
[36]**T**::= **apply ide AP**	let S:= **ide**.type in **T**.type:=find(**AP**.type,S)