Cálculo lambda — P86813_es

Olimpiada Informática Española — Final 2007 (2007)

Tu programa tiene que leer expresiones lambda y saber realizar 3 operaciones básicas del lambda cálculo: calcular las variables "gastadas", calcular las variables libres, y realizar substituciones.

Expresiones lambda. El lenguaje de las expresiones lambda (expresiones a partir de ahora) se describe con los siguientes tres casos y con ninguno más:

- Cadenas de caracteres de la `'a'` a la `'z'` son expresiones (variables).
- Dada una expresión B y una variable v, la cadena formada por el caracter `'\'`, seguido de v, seguido del caracter `'.'` seguido de B ($\backslash v\,.\,B$) es una expresión (abstracción).
- Dadas dos expresiones F y A, la cadena formada por el caracter `'('`, seguido de F, seguido del caracter `' '` (blanco), seguido de A, seguido del caracter `')'` (`(`F A`)`) es una expresión (aplicación).

Variables gastadas. El conjunto de variables gastadas $VG(E)$ de una expresión E es el conjunto de todas las variables de la `'a'` a la `'z'` que aparecen en la expresión E, de cualquier forma. Equivalentemente,

$$VG(E) = \begin{cases} \{v\} & \textbf{Si } E = v \\ VG(B) \cup \{v\} & \textbf{Si } E = \backslash v\,.\,B \\ VG(F) \cup VG(A) & \textbf{Si } E = (F\ \ A) \end{cases}$$

Variables libres. El conjunto de variables libres $VL(E)$ de una expresión E se define de esta forma:

$$VL(E) = \begin{cases} \{v\} & \textbf{Si } E = v \\ VL(B) - \{v\} & \textbf{Si } E = \backslash v\,.\,B \\ VL(F) \cup VL(A) & \textbf{Si } E = (F\ \ A) \end{cases}$$

Substitución. La substitución $E[x := E']$ de una variable v en una expresión E por otra expresión E' se define de la siguiente forma:

$$E[x := E'] = \begin{cases} v & \textbf{Si } E \text{ es una variable } v \text{ distinta de } x \\ E' & \textbf{Si } E = x \\ \backslash v\,.\,B & \textbf{Si } E = \backslash v\,.\,B \text{ y } v = x \\ \backslash v\,.\,B[x := E'] & \textbf{Si } E = \backslash v\,.\,B, v \neq x \text{ y } v \notin VL(E') \\ (F[x := E']\ \ A[x := E']) & \textbf{Si } E = (F\ \ A) \end{cases}$$

Observad que, cuando E es de la forma $\backslash v\,.\,B$, no siempre es posible realizar la substitución. Esto siempre puede corregirse aplicando una conversión alpha, descritas a continuación, en la expresión $\backslash v\,.\,B$ reemplazando v por una variable no gastada. *Sin embargo, las entradas del problema serán tales que nunca os va a ser necesario hacer una conversión alpha para hacer la substitución.*

Entrada

Un juego de pruebas es una secuencia de cálculos, cada uno de los cuales ocupa una línea. Los cálculos pueden ser de la forma "`G` E", "`L` E" o "`S` x E E'", donde E y E' son expresiones válidas y x es una variable. Ninguna de las líneas ocupará más de 2000 caracteres. En los cálculos de tipo `S` se os garantiza que la substitución puede hacerse, sin que sea necesario que hagáis ninguna conversión alpha en ningún momento.

Salida

Por cada cálculo, tu programa debe escribir una línea con la respuesta. Escribe las variables gastadas $VG(E)$ de E si la petición es de tipo `G`, las variables libres $VL(E)$ de E si es de tipo `L`, y la substitución $E[x := E']$ si la petición es de tipo `S`. Escribe un conjunto de variables como una secuencia ordenada alfabéticamente de las mismas (fíjate en los ejemplos). Observa que el resultado de la substitución puede ser más largo de 2000 caracteres.

Puntuación

- **TestA:** **25 Puntos**

  Pruebas de no más de 200 expresiones que sólo contienen peticiones de cálculo de tipo `G`.

- **TestA:** **40 Puntos**

  Pruebas de no más de 200 expresiones que sólo contienen peticiones de cálculo de tipo `L`.

- **TestA:** **35 Puntos**

  Pruebas de no más de 200 expresiones que sólo contienen peticiones de cálculo de tipo `S`.

Ejemplo de entrada 1

```
G x
G \x.(x \y.(x y))
G (x y)
G \x.(x y)
G (\x.x f)
G (\x.(x \x.x) f)
G (\y.\x.y x)
G (((\c.\t.\e.((c t) e) \a.\b.a) a) b)
```

Ejemplo de salida 1

```
x
xy
xy
xy
fx
fx
xy
abcet
```

Ejemplo de entrada 2

```
L x
L \x.(x \y.(x y))
L (x y)
L \x.(x y)
L (\x.x f)
L (\x.(x \x.x) f)
L (\y.\x.y x)
L (((\c.\t.\e.((c t) e) \a.\b.a) a) b)
```

Ejemplo de salida 2

```
x

xy
y
f
f
x
ab
```

Ejemplo de entrada 3

```
S x x y
S y y y
S x x (x z)
S x (x x) (x z)
S x (x (y x)) (\x.(f f) g)
S x \x.x (a a)
S x \y.x (a a)
S x \y.(x \x.(x x)) (a z)
```

Ejemplo de salida 3

```
y
y
(x z)
((x z) (x z))
((\x.(f f) g) (y (\x.(f f) g)))
\x.x
\y.(a a)
\y.((a z) \x.(x x))
```

Observación (no relacionada con el problema)

Si sois capaces de resolver este problema, estáis muy cerca de poder *evaluar* expresiones lambda. A continuación describimos los pasos necesarios.

Conversión alpha. El cambio en el nombre de la variable v en expresiones de la forma $\backslash v\,.\,B$ crea expresiones equivalentes.

Reducciones beta. Cualquier expresión de la forma $(\backslash x\,.\,B\ \ A)$ se reduce a $B[x := A]$, donde puede haber sido necesario aplicar una conversión alpha sobre $\backslash x\,.\,B$ renombrando x en una variable no gastada.

Forma normal. Una expresión se dice que está en forma normal cuando no pueden aplicarse más reducciones beta a ninguna de sus subexpresiones.

Información del problema

Autor : Ángel Herranz
Generación : 2014-01-29 12:57:21