

INSTITUTO TECNOLÓGICO DE COSTA RICA

TAREA 3 - OZ (MOZART)

Ariel Herrera

Saúl Zamora

profesor

M. Sc. Saúl Calderón Ramírez

September 19, 2016

I. DATOS HISTÓRICOS

Oz es un lenguaje multiparadigma desarrollado en el Laboratorio de Sistemas de Programación de la Universidad Católica de Louvain con propósitos educativos. El lenguaje fue inicialmente diseñado por Gert Smolka y sus estudiantes en 1991. Para 1996 el desarrollo de Oz continuaba en cooperación con el grupo de investigación de Seif Haridi y Peter Van Roy en el Instituto Sueco de Ciencias de la Computación. Desde 1999, Oz ha sido continuamente desarrollado por el grupo internacional conocido como *el Consorcio Mozart*, el cual consistía originalmente de la Universidad de Saarland, Universidad Católica de Louvain y el Instituto Sueco de Ciencias de la Computación. En 2005, la responsabilidad del desarrollo y mantenimiento de Oz se transfirió a los miembros principales del grupo, *la Junta de Mozart*, con el expreso propósito de abrir el desarrollo de Mozart a una comunidad más grande. El sistema de programación Mozart es la implementación principal de Oz. Ha sido publicado con una licencia *open source* por el Consorcio Mozart. También ha sido llevado a diferentes plataformas de Unix, FreeBSD, Linux, Windows y OS X.

II. IMPORTANCIA Y USOS

Oz fue construido principalmente con propósitos educativos ya que contiene la mayoría de los paradigmas de programación incluyendo lógico, funcional, imperativo, orientado a objetos, programación concurrente, restrictivo, distribuido, entre otros.

III. TIPOS DE DATOS

Oz es un lenguaje con muy pocos tipos de datos nativos:

- Números: punto flotante y enteros.
- Records: para agrupar data.
- Tuplas: records con características enteras en orden ascendente.
- Listas: estructuras lineales simples.

IV. EXPRESIONES

TABLE I
OPERADORES

Operador	Categoría
Nada	Nueva línea
<code>/* ... */</code>	Comentarios
<code>%</code>	Comentarios
<code>τ = <, <, >=, ></code>	Comparaciones
<code>Pow</code>	Potencia
<code>Log</code>	Logaritmo
<code>mod</code>	División modular
<code>Sqrt / Exp / Abs</code>	Raíz cuadrada / Función exponencial \ Valor Absoluto
<code>Sin / Cos / Tan</code>	Básicas trigonométricas
<code>Asin / Acos / Atan</code>	Inversas trigonométricas

V. ESTRUCTURAS DE CONTROL

TABLE II
ESTRUCTURAS DE CONTROL

Instrucción	Uso
<code>try a catch exn then ... end</code>	Atrapar excepciones
<code>raise ... end</code>	Arrojar excepciones
<code>if c then ... end</code>	<code>if_then</code>
<code>if c then b1 elseif c2 then b2 else b3 end</code>	<code>if_then_else</code>

VI. CARACTERÍSTICAS PRINCIPALES

Oz tiene semántica formal simple y una implementación eficiente. También es un lenguaje que facilita la concurrencia, similar a Erlang.

VII. CARACTERÍSTICAS DISTINTIVAS

Además de la programación multiparadigma, las mayores fortalezas de Oz están en la programación restrictiva y la programación distribuida. Debido a su diseño, Oz es capaz de implementar un modelo de programación de red transparente. Este modelo hace fácil programar aplicaciones abiertas y tolerantes de fallos con el lenguaje. Para la programación restrictiva, Oz introduce la idea de *espacios de computación*, dentro de los cuales se permite la búsqueda definida por el usuario y la distribución de estrategias ortogonal al dominio de las restricciones.

VIII. VENTAJAS Y DESVENTAJAS

A. Ventajas

- Dada su capacidad para la programación multiparadigma, resulta muy útil para propósitos educativos.

B. Desventajas

- La velocidad de ejecución de un programa producido por el compilador Mozart es muy lenta (alrededor de 50 veces más lento que el GCC para C).

IX. EJEMPLO

Implementación de la división con el algoritmo trial:

```
fun {Sieve Xs}
  case Xs of nil then nil
  [] X|Xr then Ys in
    thread Ys = {Filter Xr fun {$ Y} Y mod X \= 0 end}
    end
    X|{Sieve Ys}
  end
end
```

X. REFERENCIAS

REFERENCES

- [1] Oz (programming language) (2016). . In *Wikipedia*. Retrieved from [https://en.wikipedia.org/wiki/Oz_\(programming_language\)](https://en.wikipedia.org/wiki/Oz_(programming_language))
- [2] Syntax in oz. Retrieved September 20, 2016, from <http://rigaux.org/language-study/syntax-across-languages-pe>