

THE MULTIPARADIGM LANGUAGE G

Silvan YALCIN

Gruppe B

e-mail: silvanyalcin@hotmail.com

INHALTSÜBERSICHT

- Einführung
- Object-oriented Paradigms in G
- Funktional Paradigms in G
- Logik Paradigms in G
- Imperative-procedural Paradigms in G
- Multiparadigm Forschung
- Die Sprache G
- Die Basic Arten von G
- Eingebaute Arten
- Eingebaute Operation
- Strom Operatoren
- Strom Eigenschaften
- Ändern von Aufzählungsprotokoll
- Vorteile
- Nachteile
- Literaturliste
- Zusammenfassung

EINFÜHRUNG

- auf der Oregon Staate
- Jahr 1988
- unterstützt
- Ziel zu integrieren

OBJECT ORI.PARADIGM IN G

```
func {Simulation} time ( ) currentTime.  
func {Simulation} addEvent (event,eventTime)  
( nextEvent:=event);  
(nextEventTime:=eventTime).  
func {Simulation} proceed ( )  
(currentTime:=nextEventTime );  
me::processEvent(nextEvent).
```

FUNKTIONAL PARADIGMS IN G

```
func min(s) if (not(tail(s))) {(s)}
```

```
elif (head(s) < tail (s)) {min (head(s ); tail(s))}
```

```
else {min (tail(s))}.
```

LOGIK PARADIGMS IN G

```
aveTaxpayer := {
grossInc[?x , ?inc],
inc>2000 and inc <20000,
not(foreigner[x]),
not(spouse[x , ?p] and grossinc [p ,>3000]) ->}.
grossInc := {
grossSalary[?x , ?y],
investInc[x ,?z],
not(recPension[x,<5000])->[x,y+z]
}.
```

IMPERATIVE-PROZEDURALE PARADIGMAS IN G

```
func monteCarlo ()  
    local [winA:0 , winB:0 , sum , numberRolls:50 , total];  
    (total := numberRolls);  
    while (numberRolls > 0) do  
        (sum := random(6) + random(6));  
        if ( (sum>=7 and sum<=10) or sum =12)  
        {winA := winA + 1}  
        else  
        {winB := winB +1 };  
        (numberRolls := numberRolls-1)  
    end;  
    write(`When the dice were rolled` ``,numberRolls,``  
times:\n`);  
    write (`Side A won` , `winA,`` ``times.\n` );  
    write (Side B won` , `winB,`` ``times.\n` ).  
montecarlo().
```

MULTIPARADIGM FORSCHUNG

- **als verständliche Elemente**
- **mit bestimmte Eigenschaften**
- **Spezifischen Paradigmen**
- **Nial, produzieren**
- **Eine zukünftige Version unterstützen.**

DIE SPRACHE G

- interaktive, experimentale Sprache
- Werte verschiedener Art
- Die Sprache Icon
- Einfluss
- Die extreme Leichte und Generalisierung
- macht sie zur Grunddatentyp von multiparadigmalen Sprachen
- Der G Interpreter
- C durchgeführt.

DIE BASIC ARTEN VON G

- Die logischen Operatoren
 - And
 - or
 - not
- Die primitiven Datenstromoperatoren
 - head und tail

BEISPIEL

→ a:=10

→ if (a>5) {[a,a,a,]}

else {``a was less than 5``}.

[10,10,10]

→

EINGEBAUTE ARTEN

- *In der Sprache G ist jeder Wert ein Datenstromwert. Die Arten*
- *Stream*
- *Int*
- *Char*
- *Real*
- *String*

EINGEBAUTE OPERATION

- Relational
- Arithmetic
- Logical
- Stream

BEISPIEL

→ [local [x:10], x, [x-1]]

[10, [9]]

→ [local[x:1], x[local[x:2], x]]

[1[2]]

→

STREAM OPERATOREN

- Head
- Tail
- Concatenate
- Filter

STREAM EIGENSCHAFTEN

- Environment
- Value sequence
- Enumarition protocol

ÄNDERN VON AUFZÄHLUNGS PROTOKOLL

- Kombinierte Datenströme können auch durch Veränderung von einem Datenwertstrom mit einem der iterativen Präfixen
- While
- Foreach
- repeat

BEISPIEL

→foreach(1..3) do [1,2] end.

[1,2,1,2,1,2]

→infinite:=repeat[1,2] end.

[]

→

VORTEILE

- andere Paradigmen integrierbar
- Einfachheit
- diese Sprache unterstützt mehrere Paradigmen

NACHTEILE

- Es ist schwer Problemdomänen nachzugehen

ZUSAMMENFASSUNG

- eine experimentelle Sprache
- verschiedene Programmierparadigmen zu integrieren.
- Es müssen noch mehr Paradigmen erstellt und durchgeführt werden.
- Dann war der Versuch diese Bestandteile in linguistische Struktur von G zu einbinden.

LITERATURLISTE

- 1-The Multiparadigm Language G
Placer, J.R
- 2-
<http://fie.engrng.pitt.edu/fie99/papers/1350.pdf>



Vielen Dank für ihre
Aufmerksamkeit



This document was created with Win2PDF available at <http://www.win2pdf.com>.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.