



DEPARTAMENTO
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

Trabajo Práctico 2: Diseño

Primer cuatrimestre - 2015

Algoritmos y Estructuras de Datos II

Grupo 2

Integrante	LU	Correo electrónico
Benitez, Nelson	945/13	nelson.benitez92@gmail.com
Roizman, Violeta	273/11	violeroizman@gmail.com
Vázquez, Jérica	318/13	jesis_93@hotmail.com
Zavalla, Agustín	670/13	nkm747@gmail.com

Instancia	Docente	Nota
Primera entrega		
Segunda entrega		



Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

Ciudad Universitaria – Pabellón I (Planta Baja)

Intendente Güiraldes 2160 – C1428EGA

Ciudad Autónoma de Buenos Aires – Rep. Argentina

Tel/Fax: (+54 +11) 4576-3300

<http://www.exactas.uba.ar>

Índice

1. Módulo Sistema DCNet	2
1.1. Interfaz	2
1.2. Representación	2

1 Módulo Sistema DCNet

Una DCNet es

1.1 Interfaz

se explica con `DCNET`

géneros `dcnet`

Operaciones

EJEMPLO(`in r : restricción, in s : itConj(String)`) \longrightarrow `res : bool`

Pre $\equiv \{\text{true}\}$

Post $\equiv \{res =_{\text{obs}} (\forall c : \text{String}) \text{ esta?}(c, \text{secuSubY}(s)) \Rightarrow \text{verifica?}(c, r)\}$

Descripción: Verifica si todo String del iterador verifica en la restricción.

Complejidad: $O(R)$

Aliasing:

1.2 Representación

se representa con `city`

donde `city` es tupla \langle `Compus : arreglo(IP, itColaLogP, itColaLog),`
`PaquetesPorPrioridad : diccPref(compu, colaLogP(paquete)),`
`PaquetesPorID : diccPref(compu, colaLog(paquete)),`
`CompusPor#Envios : colaLogP(compus)` \rangle

Invariante de representación

1. si son hojas son tags
2. sino, son operadores
3. si el operador es unario, tiene solo hijo izquierdo

$\text{Rep} : \widehat{\text{restr}} \longrightarrow \text{boolean}$

$(\forall r : \widehat{\text{restr}})$

$\text{Rep}(r) \equiv \text{if } (r.\text{izq} \neq \text{NULL}) \text{ then}$

$\text{if } (r.\text{der} = \text{NULL}) \text{ then}$

$*r.\text{val} = \text{"NOT"} \wedge \text{rep}(*r.\text{izq})$

$\text{else } *r.\text{val} \in \text{Ag}(\text{"OR"}, \text{Ag}(\text{"AND"}, \text{vacío})) \wedge \text{rep}(*r.\text{izq}) \wedge \text{rep}(*r.\text{der})$

Función de abstracción

$\text{Abs} : \widehat{\text{restricción}} r \longrightarrow \widehat{\text{ab}}$

$\{\text{Rep}(r)\}$

$(\forall r : \widehat{\text{restricción}})$

$\text{Abs}(r) \equiv a : \widehat{\text{ab}} \mid$

$(*r.\text{izq} = \text{NULL} \wedge *r.\text{der} = \text{NULL} \iff \text{nil?}(a)) \vee_L$

$*r.\text{izq} = \text{izq}(a) \wedge *r.\text{der} = \text{der}(a) \wedge r.\text{val} = \text{raiz}(a)$

Algoritmos

```
IVERIFICA(in  $r$  : restricción, in  $s$  : trie( $\alpha$ ))  $\longrightarrow$   $res$  : bool
  if ( $*r.izq$ ) = NULL then                                     O(1)
     $res \leftarrow definido(r.val, s)$                          O(1)
  else
    if ( $*r.der$ ) = NULL then                                  O(1)
       $res \leftarrow \neg Verifica(*r.izq, s)$                   O( $R$ )
    else
      if ( $*r.val$ ) = "AND" then                               O(1)
         $res \leftarrow Verifica(*r.izq, s) \wedge Verifica(*r.der, s)$  O( $R + R$ )
      else
         $res \leftarrow Verifica(*r.izq, s) \vee Verifica(*r.der, s)$  O( $R + R$ )
      end if
    end if
  end if
end if
```

O(R)