

Trabajo práctico 2: Especificación Base De Datos

Normativa

Límite de entrega: Miércoles 25 de mayo *hasta las 22:00 hs.* Enviar PDF a algo2.dc+TP2@gmail.com con asunto "Grupo N" siendo N el número de grupo asignado.

Normas de entrega: Ver "Información sobre la cursada" en el sitio Web de la materia.
(<http://www.dc.uba.ar/materias/aed2/2016/1c/cursada>)

Versión: 1.1 del 11 de mayo de 2016 (ver TP2_Changelog.txt)

Especificación

1. TAD DATO

TAD TIPO es BOOL

TAD DATO

géneros dato

usa string, nat, tipo

exporta generadores, observadores básicos y otras operaciones

igualdad observacional

$$(\forall d_1, d_2 : \text{dato}) \left(d_1 =_{\text{obs}} d_2 \iff \left(\begin{array}{l} \text{tipo?}(d_1) \equiv \text{tipo?}(d_2) \wedge_{\text{L}} \\ ((\text{Nat?}(d_1) \Rightarrow_{\text{L}} \text{valorNat}(d_1) =_{\text{obs}} \text{valorNat}(d_2)) \wedge \\ (\text{String?}(d_1) \Rightarrow_{\text{L}} \text{valorStr}(d_1) =_{\text{obs}} \text{valorStr}(d_2))) \end{array} \right) \right)$$

generadores

datoString : string \longrightarrow dato

datoNat : nat \longrightarrow dato

observadores básicos

tipo? : dato \longrightarrow tipo

valorNat : dato $d \longrightarrow$ nat $\{\text{Nat?}(d)\}$

valorStr : dato $d \longrightarrow$ string $\{\text{String?}(d)\}$

otras operaciones

mismoTipo? : dato \times dato \longrightarrow bool

String? : dato \longrightarrow bool

Nat? : dato \longrightarrow bool

min : conj(dato) $cd \longrightarrow$ dato $\{\neg\emptyset?(cd)\}$

max : conj(dato) $cd \longrightarrow$ dato $\{\neg\emptyset?(cd)\}$

$\bullet \leq \bullet$: dato $d_1 \times$ dato $d_2 \longrightarrow$ bool $\{\text{mismoTipo?}(d_1, d_2)\}$

axiomas

$(\forall s: \text{string}, \forall n: \text{nat}, \forall d, d_1, d_2: \text{dato}, \text{paratodoconj}(\text{dato})cd)$

tipo?(datoNat(n)) \equiv true

tipo?(datoString(s)) \equiv false

valorNat(datoNat(n)) \equiv n

valorStr(datoString(s)) \equiv s

mismoTipo?(d_1, d_2) \equiv (Nat?(d_1) \equiv Nat?(d_2))

Nat?(d) \equiv tipo?(d)

String?(d) \equiv \neg Nat?(d)

Fin TAD

TAD CAMPO es STRING

TAD REGISTRO

TAD REGISTRO extiende a DICCIONARIO(CAMPO, DATO)

gêneros	registro
----------------	-----------------

usa string, dato, campo, dicc

exporta	otras operaciones
---------	-------------------

otras operaciones

campos	: registro	\rightarrow conj(campo)
borrar?	: registro <i>criterio</i> \times registro <i>r</i>	\rightarrow bool $\{\#campos(criterio) \equiv 1\}$
agregarCampos	: registro $r_1 \times$ registro r_2	\rightarrow registro
copiarCampos	: conj(campo) <i>cc</i> \times registro $r_1 \times$ registro r_2	\rightarrow registro $\{cc \in campos(r_2)\}$
coincideAlguno	: registro $r_1 \times$ conj(campo) <i>cc</i> \times registro r_2	\rightarrow bool $\{cc \subseteq campos(r_1) \cap campos(r_2)\}$
coincidenTodos	: registro $r_1 \times$ conj(campo) <i>cc</i> \times registro r_2	\rightarrow bool $\{cc \subseteq campos(r_1) \cap campos(r_2)\}$
enTodos	: campo <i>c</i> \times conj(registro) <i>cr</i>	\rightarrow bool
combinarTodos	: campo <i>c</i> \times registro $r_1 \times$ conj(registro) <i>cr</i>	\rightarrow conj(registro) $\{c \in campos(r_1) \wedge enTodos(c, cr)\}$

axiomas

$$(\forall c: \text{campo}, \forall v, d_1, d_2: \text{dato}, \forall r, r_1, r_2, \text{crit}: \text{registro}, \forall cc: \text{conj}(\text{campo}), \forall cr: \text{conj}(\text{registro}))$$

$\text{campos}(r)$	$\equiv \text{claves}(r)$
$\text{borrar?}(crit, r)$	$\equiv \text{coincidenTodos}(crit, \text{campos}(crit), r)$
$\text{agregarCampos}(r_1, r_2)$	$\equiv \text{copiarCampos}(\text{campos}(r_2) - \text{campos}(r_1), r_1, r_2)$
$\text{copiarCampos}(cc, r_1, r_2)$	$\equiv \text{if } \emptyset?(cc) \text{ then } r_1 \text{ else copiarCampos}(\text{sinUno}(cc), \text{definir}(\text{dameUno}(cc), \text{obtener}(\text{dameUno}(cc), r_2), r_1), r_2) \text{ fi}$
$\text{coincideAlguno}(r_1, cc, r_2)$	$\equiv \neg \emptyset?(cc) \wedge_L (\text{obtener}(\text{dameUno}(cc), r_1) = \text{obtener}(\text{dameUno}(cc), r_2)) \vee \text{coincideAlguno}(r_1, \text{sinUno}(cc), r_2)$
$\text{coincidenTodos}(r_1, cc, r_2)$	$\equiv \emptyset?(cc) \vee_L (\text{obtener}(\text{dameUno}(cc), r_1) = \text{obtener}(\text{dameUno}(cc), r_2) \wedge \text{coincidenTodos}(r_1, \text{sinUno}(cc), r_2))$
$\text{enTodos}(c, cr)$	$\equiv \emptyset?(cr) \vee_L (c \in \text{campos}(\text{dameUno}(cr)) \wedge \text{enTodos}(c, \text{sinUno}(cr)))$
$\text{combinarTodos}(c, r, cr)$	$\equiv \text{if } \emptyset?(cr) \text{ then } \emptyset$ else $\text{combinarTodos}(c, r, \text{sinUno}(cr)) \cup (\text{if } \text{obtener}(c, \text{dameUno}(cr)) = \text{obtener}(c, r) \text{ then } \{\text{agregarCampos}(r, \text{dameUno}(cr))\} \text{ else } \emptyset \text{ fi})$ fi

Fin TAD

TAD TABLA

$\text{claves}(\text{indexar}(c, t))$	$\equiv \text{claves}(t)$
$\text{indices}(\text{nuevaTabla}(n, cc, r))$	$\equiv \emptyset$
$\text{indices}(\text{agregarRegistro}(r, t))$	$\equiv \text{indices}(t)$
$\text{indices}(\text{borrarRegistro}(crit, t))$	$\equiv \text{indices}(t)$
$\text{indices}(\text{indexar}(c, t))$	$\equiv \text{Ag}(c, \text{indices}(t))$
$\text{campos}(\text{nuevaTabla}(n, cc, r))$	$\equiv \text{campos}(r)$
$\text{campos}(\text{agregarRegistro}(r, t))$	$\equiv \text{campos}(t)$
$\text{campos}(\text{borrarRegistro}(crit, t))$	$\equiv \text{campos}(t)$
$\text{campos}(\text{indexar}(c, t))$	$\equiv \text{campos}(t)$
$\text{tipoCampo}(c, \text{nuevaTabla}(n, cc, r))$	$\equiv \text{tipo?}(\text{obtener}(c, r))$
$\text{tipoCampo}(c, \text{agregarRegistro}(r, t))$	$\equiv \text{tipoCampo}(c, t)$
$\text{tipoCampo}(c, \text{borrarRegistro}(crit, t))$	$\equiv \text{tipoCampo}(c, t)$
$\text{tipoCampo}(c, \text{indexar}(c, t))$	$\equiv \text{tipoCampo}(c, t)$
$\text{registros}(\text{nuevaTabla}(n, cc, r))$	$\equiv \emptyset$
$\text{registros}(\text{agregarRegistro}(r, t))$	$\equiv \text{Ag}(r, \text{registros}(t))$
$\text{registros}(\text{borrarRegistro}(crit, t))$	$\equiv \text{registros}(t) - \text{coincidencias}(crit, \text{registros}(t))$
$\text{registros}(\text{indexar}(c, t))$	$\equiv \text{registros}(t)$
$\text{cantidadDeAccesos}(\text{nuevaTabla}(n, cc, r))$	$\equiv 0$
$\text{cantidadDeAccesos}(\text{agregarRegistro}(r, t))$	$\equiv 1 + \text{cantidadDeAccesos}(t)$
$\text{cantidadDeAccesos}(\text{borrarRegistro}(crit, t))$	$\equiv \# \text{coincidencias}(crit, \text{registros}(t)) + \text{cantidadDeAccesos}(t)$
$\text{cantidadDeAccesos}(\text{indexar}(c, t))$	$\equiv \text{cantidadDeAccesos}(t)$
$\text{puedoInsertar?}(r, t)$	$\equiv \text{compatible}(r, t) \wedge \neg \text{hayCoincidencia}(r, \text{claves}(t), \text{registros}(t))$
$\text{compatible}(r, t)$	$\equiv \text{campos}(r) = \text{campos}(t) \wedge_L \text{mismosTipos}(r, t)$
$\text{puedeIndexar}(c, t)$	$\equiv c \in \text{campos}(t) \wedge_L c \notin \text{indices}(t) \wedge (\# \text{indices}(t) \leq 1 \wedge (\# \text{indices}(t) = 1 \Rightarrow_L \neg \text{tipoCampo}(c, t) \equiv \text{tipoCampo}(\text{dameUno}(\text{indices}(t)), t)))$
$\text{combinarRegistros}(c, cr_1, cr_2)$	$\equiv \text{if } \emptyset?(cr_1) \text{ then } \emptyset \text{ else } \text{combinarTodos}(c, \text{dameUno}(cr_1), cr_2) \cup \text{combinarRegistros}(c, \text{sinUno}(cr_1), t_2) \text{ fi}$
$\text{hayCoincidencia}(r, cc, cr)$	$\equiv \neg \emptyset?(cr) \wedge_L (\text{coincideAlguno}(r, cc, \text{dameUno}(cr)) \vee \text{hayCoincidencia?}(r, cc, \text{sinUno}(cr)))$
$\text{coincidencias}(crit, cr)$	$\equiv \text{if } \emptyset?(cr) \text{ then } \emptyset$ else $\text{if } \text{coincidenTodos}(crit, \text{campos}(crit), \text{dameUno}(cr)) \text{ then}$ $\text{Ag}(\text{dameUno}(cr), \text{coincidencias}(crit, \text{sinUno}(cr)))$ else $\text{coincidencias}(crit, \text{sinUno}(cr))$ fi
$\text{minimo}(c, t)$	$\equiv \text{min}(\text{dameColumna}(c, \text{registros}(t)))$
$\text{maximo}(c, t)$	$\equiv \text{max}(\text{dameColumna}(c, \text{registros}(t)))$
$\text{dameColumna}(c, cr)$	$\equiv \text{if } \emptyset?(cr) \text{ then } \emptyset \text{ else } (\text{if } c \in \text{campos}(\text{dameUno}(cr)) \text{ then } \{ \text{obtener}(c, \text{dameUno}(cr)) \} \text{ else } \emptyset \text{ fi}) \cup \text{dameColumna}(c, \text{sinUno}(cr))$
$\text{mismosTipos}(r, t)$	$\equiv \emptyset?(r) \vee_L (\text{mismosTipos}(\text{borrar}(\text{dameUno}(\text{campos}(r)), r), t) \wedge \text{tipo?}(\text{obtener}(\text{dameUno}(\text{campos}(r_1)), r_1) \equiv \text{tipoCampo}(\text{dameUno}(\text{campos}(r_1)), t))$

Fin TAD

TAD BASED DATOS

hayJoin?(t_1, t_2 , agregarTabla(tbl, db))	\equiv hayJoin?(t_1, t_2, db)
hayJoin?(t_1, t_2 , insertarEntrada(reg, t_3, db))	\equiv hayJoin?(t_1, t_2, db)
hayJoin?(t_1, t_2 , borrar(cr, c, db))	\equiv hayJoin?(t_1, t_2, db)
hayJoin?(t_1, t_2 , generarVistaJoin(t_3, t_4, c_2, db))	$\equiv (t_1 = t_3 \wedge t_2 = t_4) \vee$ hayJoin?(t_1, t_2, db)
hayJoin?(t_1, t_2 , borrarJoin(t_3, t_4, c_2, db))	$\equiv \neg(t_1 = t_3 \wedge t_2 = t_4) \wedge$ hayJoin?(t_1, t_2, db)
campoJoin(t_1, t_2 , agregarTabla(tbl, db))	\equiv campoJoin(t_1, t_2, db)
campoJoin(t_1, t_2 , insertarEntrada(reg, t, db))	\equiv campoJoin(t_1, t_2, db)
campoJoin(t_1, t_2 , borrar(cr, t, db))	\equiv campoJoin(t_1, t_2, db)
campoJoin(t_1, t_2 , generarVistaJoin(t_3, t_4, c, db))	\equiv if $t_1 = t_3 \wedge t_2 = t_4$ then <div style="margin-left: 100px;">c</div> <div style="margin-left: 100px;">else</div> <div style="margin-left: 100px;">campoJoin(t_1, t_2, db)</div> <div style="margin-left: 100px;">fi</div>
campoJoin(t_1, t_2 , borrarJoin(t_3, t_4, db))	\equiv campoJoin(t_1, t_2, db)
registros(t, db)	\equiv registros(dameTabla(t, db))
cantidadDeAccesos(t, db)	\equiv cantidadDeAccesos(dameTabla(t, db))
tablaMaxima(db)	\equiv dameTabla(encontrarMaximo(dameUno(tablas(db)), sinUno(tablas(db)), db), db)
encontrarMaximo(t, ct, db)	\equiv if $\emptyset?(ct)$ then <div style="margin-left: 100px;">t</div> <div style="margin-left: 100px;">else</div> <div style="margin-left: 100px;">if cantidadDeAccesos(t, db) \geq cantidadDeAccesos(dameUno(ct), db)</div> <div style="margin-left: 100px;">then</div> <div style="margin-left: 100px;">encontrarMaximo(t, sinUno(ct), db)</div> <div style="margin-left: 100px;">else</div> <div style="margin-left: 100px;">encontrarMaximo(dameUno(ct), sinUno(ct), db)</div> <div style="margin-left: 100px;">fi</div> <div style="margin-left: 100px;">fi</div>
vistaJoin(t_1, t_2, db)	\equiv combinarRegistros(campoJoin(t_1, t_2, db), registros(dameTabla(t_1, db)), registros(dameTabla(t_2, db)))
buscar(r, t, db)	\equiv coincidencias(r , registros(dameTabla(t, db)))

Fin TAD