

# Especificación “Lollapatuza”

Algoritmos y Estructura de Datos II - FCyEN - UBA

29 de mayo de 2023

## 1. Registro de Cambios

- 29 de mayo: Error de tipeo en hackear, corresponde “i” en vez de “v”.
- 29 de mayo: Corrección firma olvidarItemDeVenta.

## 2. Lollapatuza

**TAD** LOLLAPATUZA

**géneros**      lolla

**observadores básicos**

puestos                      : lolla  $l$                        $\longrightarrow$   $\text{dicc}(\text{puestoid}, \text{puesto})$   
personas                     : lolla  $l$                        $\longrightarrow$   $\text{conj}(\text{persona})$

**generadores**

crearLolla :  $\text{dicc}(\text{puestoid}, \text{puesto}) \text{ } ps \times \text{conj}(\text{persona}) \text{ } as \longrightarrow$  lolla  $\{ \text{vendenAlMismoPrecio}(\text{significados}(ps)) \wedge \text{NoVendieronAun}(\text{significados}(ps)) \wedge \neg \emptyset?(as) \wedge \neg \emptyset?(\text{claves}(ps)) \}$   
  
vender : lolla  $l \times \text{puestoid } pi \times \text{persona } a \times \text{item } i \times \text{cant } c \longrightarrow$  lolla  $\{ a \in \text{personas}(l) \wedge \text{def?}(pi, \text{puestos}(l)) \wedge_L \text{haySuficiente?}(\text{obtener}(pi, \text{puestos}(l)), i, c) \}$   
  
hackear : lolla  $l \times \text{persona } a \times \text{item } i \longrightarrow$  lolla  $\{ \text{ConsumioSinPromoEnAlgunPuesto}(l, a, i) \}$

**otras operaciones**

masGasto : lolla  $l \longrightarrow$  persona  $\{ \neg \emptyset?(personas(l)) \}$   
gastoTotal : lolla  $l \times \text{persona } a \longrightarrow$  dinero  $\{ a \in \text{personas}(l) \}$   
menorStock : lolla  $l \times \text{item } i \longrightarrow$  puestoid

**otras operaciones auxiliares**

gastoSobre :  $\text{multiconj}(\text{puesto}) \text{ } ps \times \text{persona } a \longrightarrow$  dinero  
masGastoEntre : lolla  $l \times \text{conj}(\text{persona}) \text{ } as \longrightarrow$   $\langle \text{dinero}, \text{persona} \rangle$   $\{ \neg \emptyset?(as) \wedge as \subseteq \text{personas}(l) \}$   
  
vendenAlMismoPrecio :  $\text{multiconj}(\text{puesto}) \text{ } ps \longrightarrow$  bool  
vendenMismoPrecioQue :  $\text{multiconj}(\text{puesto}) \text{ } ps \times \text{puesto } p \longrightarrow$  bool  
consumioSinPromoPuestos :  $\text{persona } a \times \text{item } i \times \text{multiconj}(\text{puesto}) \text{ } ps \longrightarrow$  bool  
algunPuestoId : lolla  $l \times \text{persona } a \times \text{item } i \longrightarrow$  puestoid  $\{ \text{ConsumioSinPromoEnAlgunPuesto}(l, a, i) \}$   
  
algunPuestoEntre :  $\text{persona } a \times \text{item } i \times \text{dicc}(\text{puestoid} \times \text{puesto}) \text{ } ps \longrightarrow$  puestoid  $\{ \text{consumioSinPromoPuestos}(a, i, \text{significados}(ps)) \}$   
menorStockEntre :  $\text{item } i \times \text{dicc}(\text{puestoid} \times \text{puesto}) \text{ } ps \longrightarrow$  puestoid  $\{ \neg \emptyset?(\text{claves}(ps)) \}$

**axiomas**

$(\forall ps: \text{dicc}(\text{puestoid}, \text{puesto}), \forall as: \text{conj}(\text{persona}), \forall l: \text{lolla}, \forall p: \text{puesto}, \forall a: \text{persona}, \forall i: \text{item}, \forall v: \text{multiconj}(\text{item}))$   
 $\text{puestos}(\text{crearLolla}(ps, as)) \equiv ps$   
 $\text{personas}(\text{crearLolla}(ps, as)) \equiv as$

```

puestos(vender( $l$ ,  $pi$ ,  $a$ ,  $i$ ,  $c$ ))  $\equiv$  definir( $pi$ , vender(obtener( $pi$ , puestos( $l$ )),  $a$ ,  $i$ ,  $c$ ), puestos( $l$ ))
personas(vender( $l$ ,  $pi$ ,  $a$ ,  $i$ ,  $c$ ))  $\equiv$  personas( $l$ )
gastoTotal( $l$ ,  $a$ )  $\equiv$  gastoSobre(significados(puestos( $l$ )),  $a$ )
gastoSobre( $ps$ ,  $a$ )  $\equiv$  if  $\emptyset?(ps)$  then 0 else gastosDe(dameUno( $ps$ ),  $a$ ) + gastoSobre(sinUno( $ps$ ),  $a$ ) fi
masGasto( $l$ )  $\equiv$   $\pi_2$ (masGastoEntre( $l$ , personas( $l$ )))
masGastoEntre( $l$ ,  $as$ )  $\equiv$  if  $\#(as) = 1 \vee_L$  gastoTotal( $l$ , dameUno( $as$ )) >  $\pi_1$ (masGastoEntre( $l$ , sinUno( $as$ ))) then
     $\langle$ gastoTotal( $l$ , dameUno( $as$ )), dameUno( $as$ )  $\rangle$ 
else
    if gastoTotal( $l$ , dameUno( $as$ )) =  $\pi_1$ (masGastoEntre( $l$ , sinUno( $as$ ))) then
        dameUno( $\{ \langle$ gastoTotal( $l$ , dameUno( $as$ )), dameUno( $as$ )  $\rangle$ , masGastoEntre( $l$ , sinUno( $as$ ))  $\}$ )
    else
        masGastoEntre( $l$ , sinUno( $as$ ))
    fi
fi
personas(hackear( $l$ ,  $a$ ,  $i$ ))  $\equiv$  personas( $l$ )
puestos(hackear( $l$ ,  $a$ ,  $i$ ))  $\equiv$  definir(algunPuestoId( $l$ ,  $a$ ,  $i$ ),
    olvidarItem(obtener(algunPuestoId( $l$ ,  $a$ ,  $i$ ), puestos( $l$ )),  $a$ ,  $i$ ),
    puestos( $l$ ))
algunPuestoId( $l$ ,  $a$ ,  $i$ )  $\equiv$  algunPuestoEntre( $a$ ,  $i$ , puestos( $l$ ))
algunPuestoEntre( $a$ ,  $i$ ,  $ps$ )  $\equiv$  if  $i \in$  menu(obtenerUno( $ps$ ))  $\wedge_L$  consumoSinPromo?(obtenerUno( $ps$ ),  $a$ ,  $i$ ) then
    dameUno(claves( $ps$ ))
else
    algunPuestoEntre( $a$ ,  $i$ , sinUnaClave( $ps$ ))
fi
vendenAlMismoPrecio( $ps$ )  $\equiv$  if  $\emptyset?(ps)$  then
    true
else
    vendenMismoPrecioQue( $ps$ , dameUno( $ps$ ))
     $\wedge$  vendenAlMismoPrecio(sinUno( $ps$ ))
fi
vendenMismoPrecioQue( $ps$ ,  $p$ )  $\equiv$  if  $\emptyset?(ps)$  then
    true
else
    vendenMismoPrecio(obtenerUno( $ps$ ),  $p$ )
     $\wedge$  vendenMismoPrecioQue(sinUno( $ps$ ),  $p$ )
fi
consumioSinPromoPuestos( $a$ ,  $i$ ,  $ps$ )  $\equiv$  if  $\emptyset?(claves(ps))$  then
    false
else
    ( $i \in$  menu(obtenerUno( $ps$ ))  $\wedge_L$  consumoSinPromo?(obtenerUno( $ps$ ),  $a$ ,  $i$ ))
     $\vee_L$  consumoSinPromoPuestos(sinUnaClave( $a$ ,  $i$ ,  $ps$ ))
fi
menorStock( $l$ ,  $i$ )  $\equiv$  menorStockEntre( $i$ , puestos( $l$ ))
menorStockEntre( $i$ ,  $ps$ )  $\equiv$  if  $\#(claves(ps)) = 1 \vee_L$  ( $i \in$  menu(obtenerUno( $ps$ ))  $\wedge_L$ 
    stock(obtenerUno( $ps$ ),  $i$ ) < stock(obtener(menorStockEntre( $i$ , sinUnaClave( $ps$ )),  $ps$ ),  $i$ ))
then
    dameUno(claves( $ps$ ))
else
    menorStockEntre( $i$ , sinUnaClave( $ps$ ))
fi

```

**Fin TAD**

## 2.1. Predicados Auxiliares

NoVendieronAun( $ps$ )  $\equiv$  ( $\forall a : persona$ ) ( $\forall p : puesto$ ) ( $p \in$  significados( $ps$ )  $\Rightarrow_L \emptyset?(ventas(p, a))$ )

ConsumioSinPromoEnAlgunPuesto( $l$ ,  $a$ ,  $i$ )  $\equiv$   $a \in$  personas( $l$ )

$\wedge_L$  ( $\exists pi : puestoid$ ) ( $def?(pi, puestos(l)) \wedge_L$  consumoSinPromo?(obtener( $pi$ , puestos( $l$ )),  $a$ ,  $i$ ))

### 3. PuestoDeComida

#### TAD PUESTODECOMIDA

**géneros**      puesto

#### observadores básicos

menu	: puesto $p$	$\longrightarrow$ conj(item)	
precio	: puesto $p \times$ item $i$	$\longrightarrow$ dinero	$\{i \in \text{menu}(p)\}$
stock	: puesto $p \times$ item $i$	$\longrightarrow$ cant	$\{i \in \text{menu}(p)\}$
descuento	: puesto $p \times$ item $i \times$ cant $c$	$\longrightarrow$ nat	$\{i \in \text{menu}(p)\}$
ventas	: puesto $p \times$ persona $a$	$\longrightarrow$ multiconj( $\langle$ item, cant $\rangle$ )	

#### generadores

crearPuesto	: dicc(item, nat) $p \times$ dicc(item, nat) $s \times$ dicc(item, dicc(cant, nat)) $d$	$\longrightarrow$ puesto	$\{\text{claves}(p) = \text{claves}(s) \wedge \text{claves}(d) \subseteq \text{claves}(p)\}$
vender	: puesto $p \times$ persona $a \times$ item $i \times$ cant $c$	$\longrightarrow$ puesto	$\{\text{haySuficiente?}(p, i, c)\}$

#### otras operaciones

haySuficiente?	: puesto $p \times$ item $i \times$ cant $c$	$\longrightarrow$ bool	$\{i \in \text{menu}(p)\}$
consumioSinPromo?	: puesto $p \times$ persona $a \times$ item $i$	$\longrightarrow$ bool	$\{i \in \text{menu}(p)\}$
gastosDe	: puesto $p \times$ persona $a$	$\longrightarrow$ dinero	
olvidarItem	: puesto $p \times$ persona $a \times$ item $i$	$\longrightarrow$ puesto	$\{i \in \text{menu}(p) \wedge_L \text{consumioSinPromo?}(p, a, i)\}$
vendenMismoPrecio	: puesto $\times$ puesto	$\longrightarrow$ bool	

#### otras operaciones auxiliares

consumioSinPromoVentas?	: puesto $p \times$ item $i \times$ multiconj( $\langle$ item, cant $\rangle$ ) $m$	$\longrightarrow$ bool	$\{i \in \text{menu}(p) \wedge \text{SonVentasDelMenu}(m, p)\}$
consumioSinPromo1Venta?	: puesto $p \times$ item $i \times \langle$ item, cant $\rangle v$	$\longrightarrow$ bool	$\{i \in \text{menu}(p) \wedge \pi_1(v) \in \text{menu}(p)\}$
ventasSinPromo	: puesto $p \times$ persona $a \times$ item $i$	$\longrightarrow$ multiconj( $\langle$ item, cant $\rangle$ )	$\{i \in \text{menu}(p)\}$
ventasSinPromoEntre	: puesto $p \times$ item $i \times$ multiconj( $\langle$ item, cant $\rangle$ ) $m$	$\longrightarrow$ multiconj( $\langle$ item, cant $\rangle$ )	$\{i \in \text{menu}(p) \wedge \text{SonVentasDelMenu}(m, p)\}$
olvidarItemDeVenta	: puesto $p \times$ persona $a \times$ item $i \times \langle$ item, cant $\rangle v$	$\longrightarrow$ puesto	$\{i \in \text{menu}(p) \wedge \pi_1(v) \in \text{menu}(p) \wedge_L \text{consumioSinPromo1Venta?}(p, a, i, v)\}$
gastosDeVentas	: puesto $p \times$ multiconj( $\langle$ item, cant $\rangle$ ) $m$	$\longrightarrow$ dinero	$\{\text{SonVentasDelMenu}(m, p)\}$
gastosDe1Venta	: puesto $p \times \langle$ item, cant $\rangle v$	$\longrightarrow$ dinero	$\{\pi_1(v) \in \text{menu}(p)\}$
vendenMismoPrecioMenu	: puesto $p1 \times$ puesto $p2 \times$ conj(item) $is$	$\longrightarrow$ bool	$\{is \subseteq \text{menu}(p1) \cap \text{menu}(p2)\}$

#### axiomas

( $\forall p, p1, p2$ : puesto,  $\forall a, a1, a2$ : persona,  $\forall c$ : cant,  $\forall i$ : item,  $\forall v, v1, v2$ :  $\langle$ item, cant $\rangle$ ,  $\forall m$ : multiconj( $\langle$ item, cant $\rangle$ ),  $\forall is$ : conj(item),  $\forall s$ : dicc(item, nat),  $\forall d$ : dicc(item, dicc(cant, nat)))

menu(crearPuesto( $p, s, d$ ))  $\equiv$  claves( $p$ )

precio(crearPuesto( $p, s, d$ ),  $i$ )  $\equiv$  obtener( $i, p$ )

stock(crearPuesto( $p, s, d$ ),  $i$ )  $\equiv$  obtener( $i, s$ )

descuento(crearPuesto( $p, s, d$ ),  $i, c$ )  $\equiv$  if  $c = 0$  then

0

else

if  $\neg \text{def?}(i, d)$  then

0

else

if  $\text{def?}(c, \text{obtener}(i, d))$  then

obtener( $c, \text{obtener}(i, d)$ )

else

descuento(crearPuesto( $p, s, d$ ),  $i, c - 1$ )

fi

fi

fi

ventas(crearPuesto( $p, s, d$ ),  $a$ )  $\equiv \emptyset$

menu(vender( $p, a, i, c$ ))  $\equiv$  menu( $p$ )

```

precio(vender(p, a, i1, c), i2)  $\equiv$  precio(p, i2)
stock(vender(p, a, i1, c), i2)  $\equiv$  if  $i1 = i2$  then stock(p, i2) - c else stock(p, i2) fi
descuento(vender(p, a, i1, c1), i2, c2)  $\equiv$  descuento(p, i2, c2)
ventas(vender(p, a, i, c), a2)  $\equiv$  if  $a1 = a2$  then Ag( $\langle i, c \rangle$ , ventas(p, a2)) else ventas(p, a2) fi
haySuficiente?(p, i, c)  $\equiv$  stock(p, i)  $\geq$  c
gastosDe(p, a)  $\equiv$  gastosDeVentas(p, ventas(p, a))
gastosDeVentas(p, m)  $\equiv$  if  $\emptyset?(m)$  then 0 else gastosDeVentas(p, sinUno(m)) + gastosDe1Venta(p, dameUno(m)) fi
gastosDe1Venta(p, v)  $\equiv$  aplicarDescuento( $\pi_2(v) \times$  precio(p,  $\pi_1(v)$ ), descuento(p,  $\pi_1(v)$ ,  $\pi_2(v)$ ))
consumioSinPromo?(p, a, i)  $\equiv$  consumioSinPromoVentas?(p, i, ventas(p, a))
consumioSinPromoVentas?(p, i, m)  $\equiv$  if  $\emptyset?(m)$  then
    false
  else
    consumoSinPromo1Venta?(p, i, dameUno(m))  $\vee$ 
    consumoSinPromoVentas?(p, i, sinUno(m))
  fi
consumioSinPromo1Venta?(p, i, m)  $\equiv$  if descuento(p,  $\pi_1(v)$ ,  $\pi_2(v)$ ) = 0 then true else false fi
ventasSinPromoEntre(p, i, m)  $\equiv$  if  $\emptyset?(m)$  then
     $\emptyset$ 
  else
    if consumioSinPromo1Venta?(p, i, dameUno(m)) then
      Ag(dameUno(m), ventasSinPromoEntre(p, i, sinUno(m)))
    else
      ventasSinPromoEntre(p, i, sinUno(m))
    fi
  fi
ventasSinPromo(p, a, i)  $\equiv$  ventasSinPromoEntre(p, i, ventas(p, a))
olvidarItemDeVenta(vender(p, a1, i1, c), a2, i2, v2)  $\equiv$  if  $a1 = a2 \wedge \langle i1, c \rangle = v2$  then
    if  $c = 1$  then p else vender(p, a1, c - 1) fi
  else
    vender(olvidarItemDeVenta(p, a2, i, v2), a1, i1, c)
  fi
olvidarItem(p, a, i)  $\equiv$  olvidarItemDeVenta(p, a, i, dameUno(ventasSinPromo(p, a, i)))
vendenMismoPrecio(p1, p2)  $\equiv$  vendenMismoPrecioMenu(p1, p2, menu(p1)  $\cap$  menu(p2))
vendenMismoPrecioMenu(p1, p2, is)  $\equiv$  if  $\emptyset?(is)$  then
    true
  else
    precio(p1, dameUno(is)) = precio(p2, dameUno(is))
     $\wedge$  vendenMismoPrecioMenu(p1, p2, sinUno(is))
  fi

```

**Fin TAD**

### 3.1. Predicados Auxiliares

$\text{SonVentasDelMenu}(m, p) \equiv (\forall v : \langle item, cant \rangle)(v \in m \Rightarrow_L (\forall i' : item) \pi_1(v) \in \text{menu}(p))$

## 4. Dinero

**TAD DINERO**

**géneros**      dinero

**extiende**    NAT

**otras operaciones**

div : dinero  $n \times \text{nat } k \longrightarrow$  dinero

aplicarDescuento : dinero  $p \times \text{nat } d \longrightarrow$  dinero

$\{0 < k\}$   
 $\{d < 100\}$

**axiomas**

$(\forall n: \text{dinero}, \forall k: \text{nat})$

div(n, k)  $\equiv$  **if**  $n < k$  **then** 0 **else** 1+div(n - k, k) **fi**

$\text{aplicarDescuento}(p, d) \equiv \text{div}(p \times (100 - d), 100)$

**Fin TAD**

## 5. Otros TADs

El TAD ITEM es renombre de NAT con género *item*.

El TAD PERSONA es renombre de NAT con género *persona*. Representa el DNI de una persona.

El TAD CANTIDAD es renombre de NAT con género *cant*.

El TAD PUESTOID es renombre de NAT con género *puetoid*.

Extendemos TAD DICCIONARIO con las siguientes operaciones:

$\text{sinUnaClave} : \text{dicc}(\alpha, \beta) \ d \longrightarrow \text{dicc}(\alpha, \beta)$

$\{\neg \emptyset?(claves(d))\}$

$\text{obtenerUno} : \text{dicc}(\alpha, \beta) \ d \longrightarrow \beta$

$\{\neg \emptyset?(claves(d))\}$

$\text{significados} : \text{dicc}(\alpha, \beta) \longrightarrow \text{multiconj}(\beta)$

### **axiomas**

$\text{sinUnaClave}(d) \equiv \text{borrar}(\text{dameUno}(claves(d)), d)$

$\text{obtenerUno}(d) \equiv \text{obtener}(\text{dameUno}(claves(d)), d)$

$\text{significados}(d) \equiv \text{if } \emptyset?(claves(d)) \text{ then } \emptyset \text{ else } \text{Ag}(\text{obtenerUno}(d), \text{significados}(\text{sinUnaClave}(d))) \text{ fi}$