

TDA COLA

Ing. Mario Milton López Winnipeg

4.5 Modalidades de colas

- En esta sección mostraremos dos modalidades usuales de colas:
 - *Dicolas*
 - *Colas de Prioridad*

4.5.1 Dicolos

Las *Dicolos* son casos particulares de listas y generalizaciones de colas en donde las eliminaciones e inserciones pueden realizarse en ambos extremos de la lista.

El conjunto de operaciones de una dicola consiste en las operaciones de una cola junto con las siguientes:

```
public Elemento ultimo();  
public void poner_frente(Objecto elemento);  
public void sacar_final(Objecto Elemento)
```

4.5.1 Especificación del TDA DiCola

Especificación Formal

Tipo: DiCola (Elemento)

□ *Sintaxis:*

- `crea` → Cola
- `vacía(Cola)` → booleano
- `primero(Cola)` → Elemento
- `poner(Cola,Elemento)` → DiCola //Poner_inicio
- `sacar(Cola, Elemento)` → DiCola // Sacar Inicio
- `Poner_Frente(Cola,Elemento)` → DiCola
- `Sacar_Final(Cola, Elemento)` → DiCola
- `Ultimo(Cola)` → Elemento

4.5.1 Especificación del TDA DiCola



88,20,30,40,50

L.Poner(60)

L.Sacar(e)

E=10

L.Poner_Frente(88)

L.Sacar_Final(e)

E=60

E=L.primer()

E=L.ultimo()

4.5.2 Cola de Prioridad

Una cola de prioridad es una cola a cuyos elementos se les ha asignado una prioridad, de forma que el orden en que los elementos son procesados sigue las siguientes reglas:

- ☐ El elemento con mayor prioridad es procesado primero.
- ☐ Dos elementos con la misma prioridad son procesados según el orden en que fueron introducidos en la cola

4.5.2. Cola de Prioridad

La interface del TDA Cola de Prioridad puede ser la siguiente:

```
publico interface ColaPrioridad
inicio
    publico boolean vacia();
    publico Object primero();
    publico void poner( elemento E, entero prioridad)
    publico void frecuencia_Prioridad ( entero frec,
        entero prioridad)
    public void sacar(Elemento E );
Fin // fin interface ColaPrioridad
```

4.5.2 Implementacion con vectores

Definición básica de la clase Cola de Prioridad representada con Vectores

Clase ColaPri

Atributos

VC arreglo(MAX) Ccola
VF arreglo[MAX] numero
colaAct numero
cant numero

Metodos

Crear()
vacía();
primero();
poner(elemento E, entero prioridad)
frecuencia_Prioridad (entero frec, entero prioridad)
sacar(Elemento E);

Fin

Constructor ColaPri.Crear

inicio
// Crear Objetos vc[1], vc[2], vc[3] ..
fin

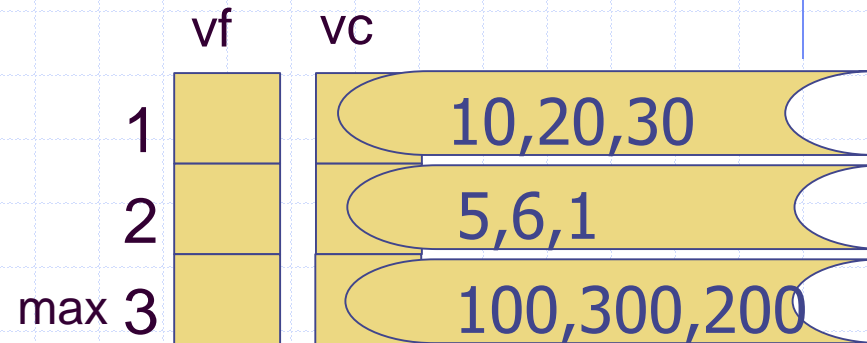
ColaPri.Poner(E: Elemento , pri :numero)

inicio
vc[pri].poner(E);

Fin

ColaAct Cant

1 0



1 Tercera edad
2 Vip
3 Regular

4.5.2 Implementación con vectores

ColaPri.Frecuencia prioridad(frec,prio)

inicio

vf[prio]= frec

Fin

ColaAct Cant

1

0

	vf	vc
1	1	10,20,30
2	2	5,6,1
max 3	1	100,300,200

- 1 Tercera edad
- 2 Vip
- 3 Regular

4.5.2 Implementación con vectores

ColaPri.Sacar(ES E: Elemento)

inicio

```
si no vc[ColaAct].vacía()
    entonces
        si cant < vf[colaAct]
            cant = cant +1
            vc[ColaAct].sacar(e)
            si cant=vf[colaAct]
                entonces
                    colaAct = colaAct +1
                    cant = 0
//retornar e
```

Fin

ColaAct Cant
2 0

E=100

Atendidos =10,5,6,100

