

Implementación del TAD Cola

La representación dinámica del TAD *cola* se puede realizar de forma análoga a la del TAD *pila*, es decir, encadenando mediante punteros los elementos de la cola.

Para que la operación de añadir un elemento a la cola sea de orden $O(1)$, se utiliza una variable puntero adicional que apunta siempre al último elemento añadido.

Las estructuras de datos a usar son:

tipo

punteroDato = puntero a unDato

unDato = registro

dato: elemento;

sig: punteroDato;

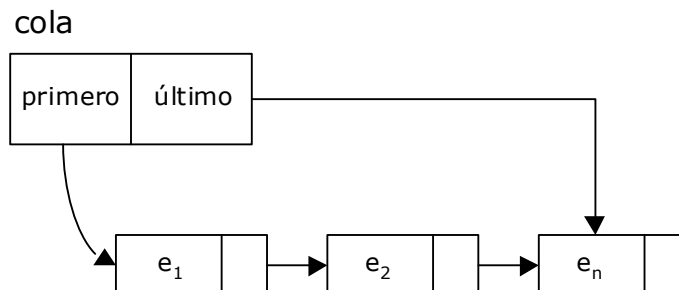
fregistro;

cola = registro

primero, ultimo: punteroDato;

fregistro;

Gráficamente, la representación de una cola, utilizando la implementación dinámica es la siguiente:



La implementación dinámica del TAD *cola*, con las operaciones de asignación y liberación, cuya necesidad en el caso de usar representación dinámica se explicó en el TAD *Pila*, es:

módulo colas

importa defTipoElemento

exporta (* La documentación del módulo de definición se deja como ejercicio para el alumno *)

tipo cola

acción crearCola (**var** c:cola)

acción añadir (**var** c:cola; e:elemento)

acción eliminar (**var** c:cola)

función primero (c:cola): elemento

función esVacía (c:cola): booleano

acción asignar (**var** nueva:cola; vieja:cola)

acción liberar (**var** c:cola)

implementación

tipo

punteroDato = **puntero a** unDato

undato = **registro**

dato: elemento;

sig: punteroDato;

fregistro;

cola = **registro**

primero, ultimo: punteroDato;

fregistro;

acción crearCola (**var** c:cola);

c.primero:= nulo;

c.ultimo:= nulo;

facción

acción añadir (**var** c:cola;e:elemento);

(* Se añade por el final de la cola *)

var aux: punterodato; **fvar**

si c.primero = nulo **entonces**

reservar (c.primero);

c.ultimo:= c.primero

sino

reservar (c.ultimo^.sig);

c.ultimo:= c.ultimo^.sig;

fsi

c.ultimo^.dato:= e;

c.ultimo^.sig:= nulo;

facción

acción eliminar (**var** c:cola);
(* Se elimina por el frente de la cola *)

var aux: punterodato; **fvar**

aux:= c.primeros;
c.primeros:= c.primeros^.sig;
liberar(aux)

facción

función primero (c:cola): elemento;
retorna (c.primeros^.dato);
ffunción

función esVacía (c:cola): booleano;
retorna (c.primeros=nulo);
ffunción

acción asignar (**var** nueva:cola; vieja:cola);

si esVacía(vieja) **entonces** creaVacía(nueva)
sino
reservar (nueva.primeros);
nueva.primeros^.dato:= vieja.primeros^.dato;
nueva.ultimo:= nueva.primeros;
vieja.primeros:= vieja.primeros^.sig;
mientras vieja.primeros <> nulo **hacer**
reservar (nueva.ultimo^.sig);
nueva.ultimo:= nueva.ultimo^.sig;
nueva.ultimo^.dato:= vieja.primeros^.dato;
vieja.primeros:= vieja.primeros^.sig;
fmientras
nueva.ultimo^.sig:= nulo;

fsi
facción

acción liberar (**var** c:cola);

c.ultimo:= c.primeros;
mientras c.primeros <> nulo **hacer**
c.primeros:= c.primeros^.sig;
liberar (c.ultimo);
c.ultimo:= c.primeros;

fmientras
facción

fmódulo