

Universidad de Guanajuato División de Ingenierías
Campus Irapuato Salamanca (DICIS)

Algoritmos y estructura de datos
Carlos Hugo García Capulín

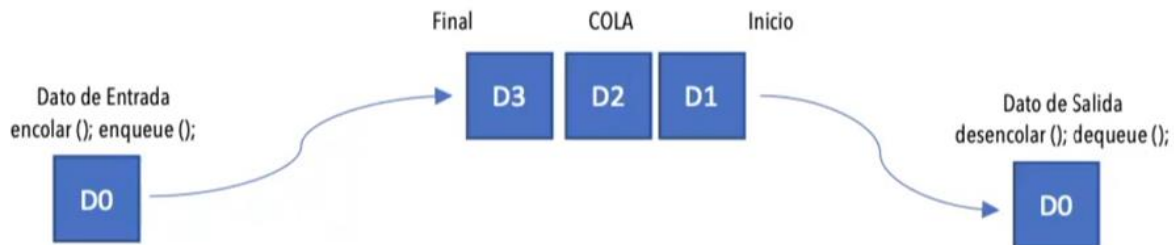
Tarea No. 10
Reporte Estructura Fila

Jair Chávez Islas
28/Octubre/2021

Problema

Una cola es un tipo de dato abstracto del tipo FIFO (First-In First-Out), es decir, el primer dato en entrar es el primero en salir, su finalidad es para almacenar datos en espera a ser procesados conforme llegan o se hacen las solicitudes de procesamiento.

La forma más simple de implementar esta estructura es considerando que los datos estarán siendo agrupados en forma de un arreglo:



Además de este tipo de estructura “cola” existen algunas variantes denominadas:

- Colas circulares. Es una cola en las que el último elemento y el primero están unidos.
- Colas de prioridad. Es una cola en la cual se tiene un orden de prioridad especial de salida de los datos, cuando más de un dato tiene la misma prioridad entonces se utiliza el principio FIFO.
- Colas doblemente terminadas. Son colas donde los elementos pueden entrar o salir, ya sea por el principio o por el final de la cola.

Las operaciones básicas relacionadas con la estructura cola son:

- | | |
|---------------------------------|------------------------------|
| 1. Insertar elemento en la cola | <code>enqueue();</code> |
| 2. Sacar un elemento de la cola | <code>dequeue();</code> |
| 3. Tamaño de la cola | <code>sizeofqueue();</code> |
| 4. Elemento Inicial | <code>frontofqueue();</code> |
| 5. Cola vacía | <code>isEmpty();</code> |
| 6. Cola llena | <code>isFull();</code> |

Continuando con el programa que comenzamos a hacer en clase con el maestro, tenemos la mayor parte del programa faltando solo 4 funciones Dequeue, quita el primer valor de la fila, isEmpty que nos avisa cuando la fila está vacía, frontofqueue que nos da el primer valor de la fila y sizeofqueue que nos da el número de valores que hay en la fila.

Teniendo el ejemplo de que hagamos una pila de 5 datos que sean los siguientes 10,20,30,40,50

Quedarían impresos de la siguiente manera:

50
40
30

20

10

El primer dato de la fila es el número 10, al eliminar un dato con la función dequeue, se iría el mismo 10 ya que es el primer elemento que ingresamos, entonces al volver a imprimir los datos quedaría de la siguiente manera

50

40

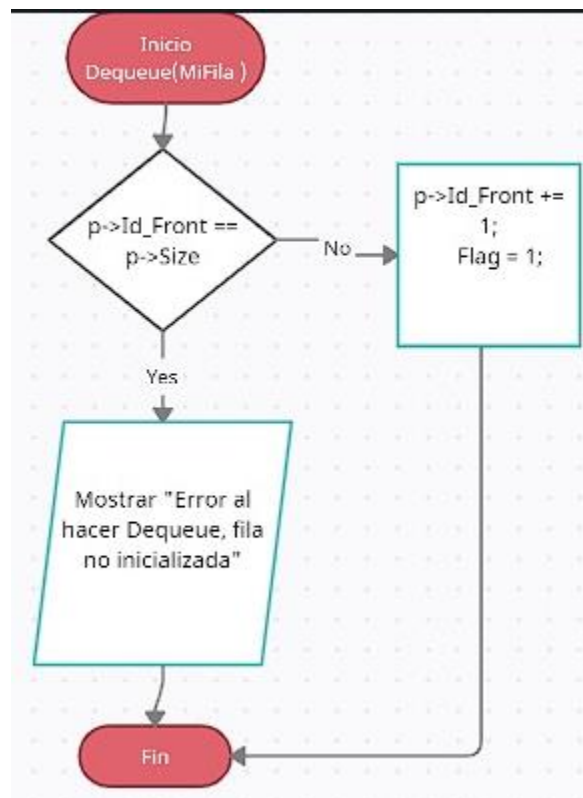
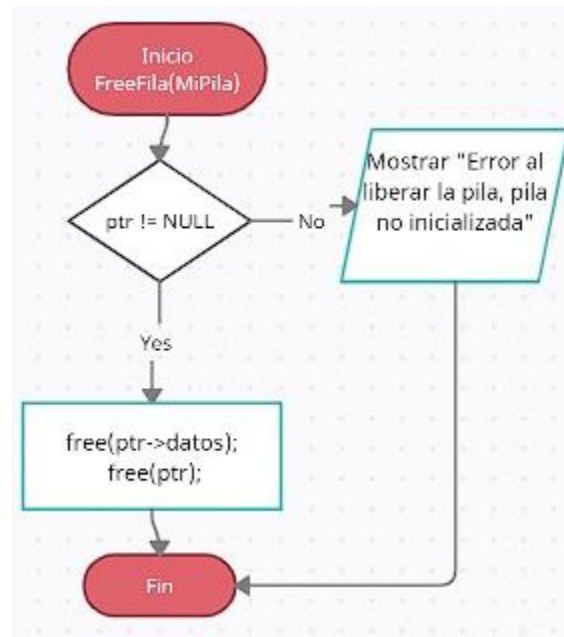
30

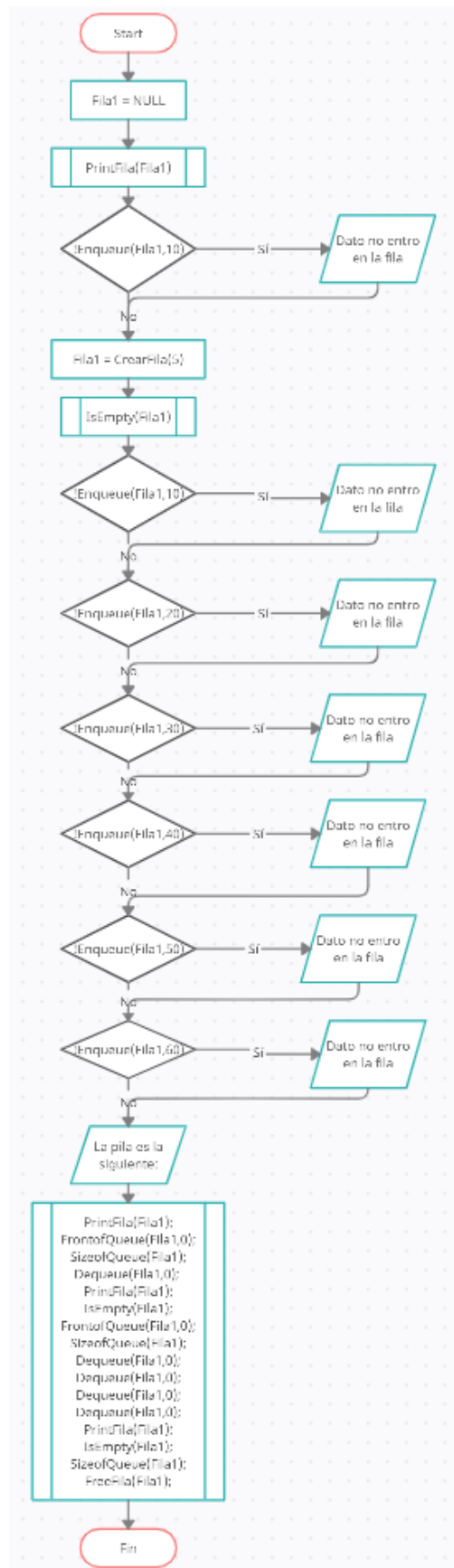
20

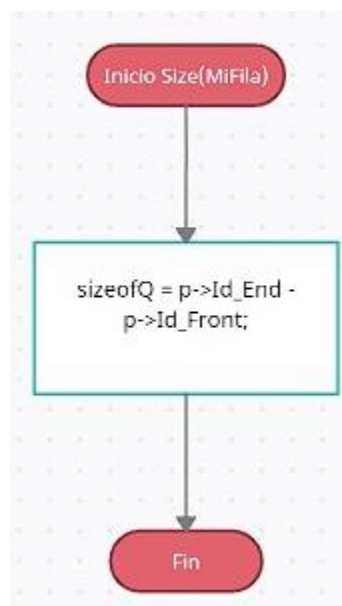
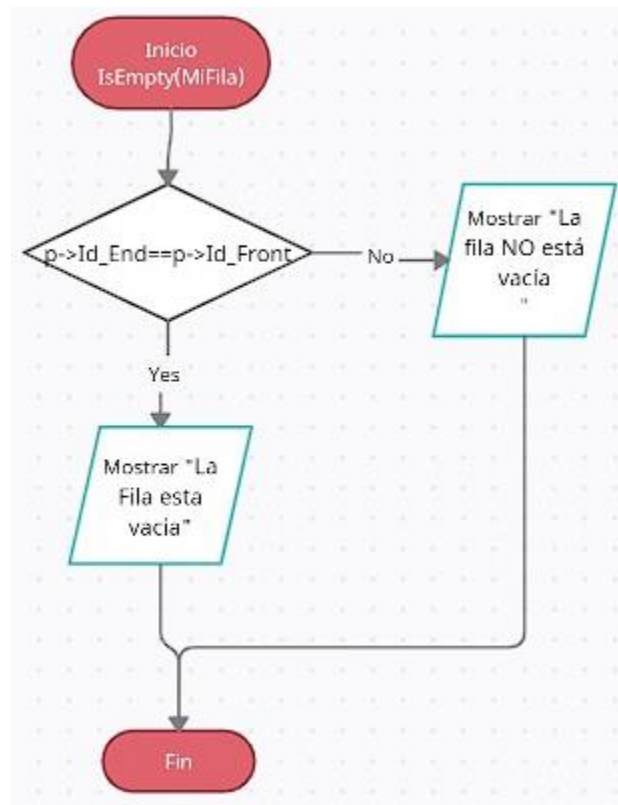
Pero aún tenemos nuestra pila con datos, por lo tanto, no está vacía, y ahora nuestro primer dato en la fila es el número 20 pero ya que hayamos eliminado con la función otros 4 datos, quedaría nuestra pila vacía así que deberíamos tener un mensaje para cuando eso pase.

Solución implementada

Diagrama del programa







Código comentado del programa

```

1 //Agregamos las librerias necesarias para las funciones que necesitamos
2 #include <stdio.h>
3 #include <stdlib.h>
4
5 //Declaramos una estructura
6 typedef struct
7 {
8     //Miembros de la estructura
9     int *data;
10    unsigned int Id_Front;
11    unsigned int Id_End;
12    unsigned int Size;
13 }FILA;//Nombre de la estructura
14
15 //Aqui ponemos los prototipos de las funciones utilizadas
16 FILA* CrearFila(unsigned int Tam);
17 void FreeFila(FILA *pf);
18 void PrintFila(FILA *pf);
19 char Enqueue(FILA *pf, int dato);
20 char IsFull(FILA *pf);
21 char Dequeue(FILA *p, int *dato);
22 char IsEmpty(FILA *p);
23 char FrontofQueue(FILA *p,int*dato);
24 unsigned int SizeofQueue(FILA *p);
25
26 //Inicializamos la funcion principal
27 int main()
28 {
29     //Declaracion de variables de esta funcion
30     FILA *Fila1=NULL;//Declaramos la variable MiPila de tipo FILA
31
32     PrintFila(Fila1);//Imprimimos la pila
33
34     if(!Enqueue(Fila1,10))//aplicamos la funcion enqueue a fila1, con elv alor de 10
35         printf("\n Dato no entro en la fila.");
36     Fila1 = CrearFila(5);//Guardamos en Fila1 la funcion crearFila con 6 elementos
37     IsEmpty(Fila1);
38     //aplicamos la funcion enqueue a fila1, con elv alor de 10
39     if(!Enqueue(Fila1,10))
40         printf("\n Dato no entro en la fila.");
41     //aplicamos la funcion enqueue a fila1, con elv alor de 20
42     if(!Enqueue(Fila1,20))
43         printf("\n Dato no entro en la fila.");
44     //aplicamos la funcion enqueue a fila1, con elv alor de 30
45     if(!Enqueue(Fila1,30))
46         printf("\n Dato no entro en la fila.");
47     //aplicamos la funcion enqueue a fila1, con elv alor de 40
48     if(!Enqueue(Fila1,40))
49         printf("\n Dato no entro en la fila.");
50     //aplicamos la funcion enqueue a fila1, con elv alor de 50
51     if(!Enqueue(Fila1,50))

```

```

50 //aplicamos la funcion enqueue a fila1, con elv alor de 50
51 if(!Enqueue(Fila1,50))
52     printf("\n Dato no entro en la fila.");
53 //aplicamos la funcion enqueue a fila1, con elv alor de 60
54 if(!Enqueue(Fila1,60))
55     printf("\n Dato no entro en la fila.");
56 printf("\nLa fila es al siguiente: \n");
57
58 //Imprimimos la fila con los valores agregados
59 PrintFila(Fila1);
60 //Mostramos el primer valor de la fila
61 FrontofQueue(Fila1,0);
62 //Mostramos la cantidad de datos de la fila
63 SizeofQueue(Fila1);
64 //Eliminamos el primer valor de la fila
65 Dequeue(Fila1,0);
66 //Se imprime la fila con el valor eliminado
67 PrintFila(Fila1);
68 //Verificamos si la fila esta vacia
69 IsEmpty(Fila1);
70 //Con un valor menos, volvemos a imprimir el valor
71 FrontofQueue(Fila1,0);
72 //Con un valor menos, volvemos a mostrar la cantidad de datos de la fila
73 SizeofQueue(Fila1);
74 //Eliminamos el primer valor de la fila
75 Dequeue(Fila1,0);
76 //Eliminamos el primer valor de la fila
77 Dequeue(Fila1,0);
78 //Eliminamos el primer valor de la fila
79 Dequeue(Fila1,0);
80 //Eliminamos el primer valor de la fila
81 Dequeue(Fila1,0);
82 //Se imprime la fila con los valores eliminados
83 PrintFila(Fila1);
84 //Comprobamos si la fila esta vacia
85 IsEmpty(Fila1);
86 //Mostramos la cantidad de datos de la fila
87 SizeofQueue(Fila1);
88
89 //Liberamos la memoria de la fila
90 FreeFila(Fila1);
91
92 //Se imprime un salto de linea
93 printf("\n");
94 return 0;//se retorna el valor de 0 para verificar que haya finalizado sin problemas
95 }
96
97 //Se inicializa la funcion dequeue
98 char Dequeue(FILA *p, int *dato)
99 {

```



```

100 //Declaracion de varibales de la funcion
101 char Flag;
102
103 //Se comprueba si la fila existe
104 if(p->Id_Front == p->Size)
105 {
106     printf("\nError al hacer el dequeue, fila no inicializada");
107     Flag = 0;
108 }
109 else
110 {
111     //Elimina el ultimo primer valor de la fila
112     p->Id_Front += 1;
113     Flag = 1;
114 }
115 printf("\ndespues de hacer el dequeue: ");
116 return Flag;
117 }
118
119 //Se inicializa la funcion IsEmpty
120 char IsEmpty(FILA *p)
121 {
122     char Flag = 0;
123     //En caso de que la pila este vacia se muestra el mensaje
124     if(p->Id_End==p->Id_Front)
125     {
126         printf("\n La Fila essta vacia!");
127         return Flag;
128     }
129     else
130     {
131         //De ser la contrario muestra este otro mensaje
132         printf("\n La fila no esta vacia");
133     }
134 }
135
136 //Se inicializa la funcion frontofqueue
137 char FrontofQueue(FILA *p,int*dato)
138 {
139     //Se declaran las variables locales de la funcion
140     char Flag;
141     unsigned int frontofQ;
142
143     //Se verifica si la pila se ha inicializado
144     if(p->Id_Front == p->Size)
145     {
146         printf("\nError, fila no inicializada");
147         Flag = 0;
148     }
149     else
150     {

```

```

151     //Se le asigna a la variable frontofQ el valor del id front
152     frontofQ = p->data[p->Id_Front];
153     //Imprimimos el valor
154     printf("\nFront of queue: %i", frontofQ);
155     }
156
157     return Flag;
158 }
159
160 //Se inicializa la funcion sizeofqueue
161 unsigned int SizeofQueue(FILA *p)
162 {
163     //Se declaran las variables de la funcion
164     unsigned int sizeofQ = 0;
165     //Se guarda en la variable el id end menos el id front
166     sizeofQ = p->Id_End - p->Id_Front;
167
168     //Se imprime el valor
169     printf("\nLa fila tiene %u datos.", sizeofQ);
170     return sizeofQ;
171 }
172
173 //Se inicializa la funcion IsFull
174 char IsFull(FILA *pf)
175 {
176     //Se verifica si la fila se llena
177     if(pf->Id_End < pf->Size)
178         return 0;
179     else
180         return 1;
181 }
182
183 //Se inicializa la funcion enqueue
184 char Enqueue(FILA *pf, int dato)
185 {
186     //Se declaran las variables locales de la funcion
187     char Flag=0;
188
189     //Se verifica si la pila se ha iniciado
190     if (pf==NULL)
191     {
192         printf("Error: la pila no esta inicializada");
193     }
194     //Si la fila no esta llena...
195     else if(!IsFull(pf))
196     {
197         //... Se agrega el dato a la fila
198         pf->data[pf->Id_End]=dato;
199         //El id aumenta en 1
200         pf->Id_End++;
201         Flag = 1;

```

```

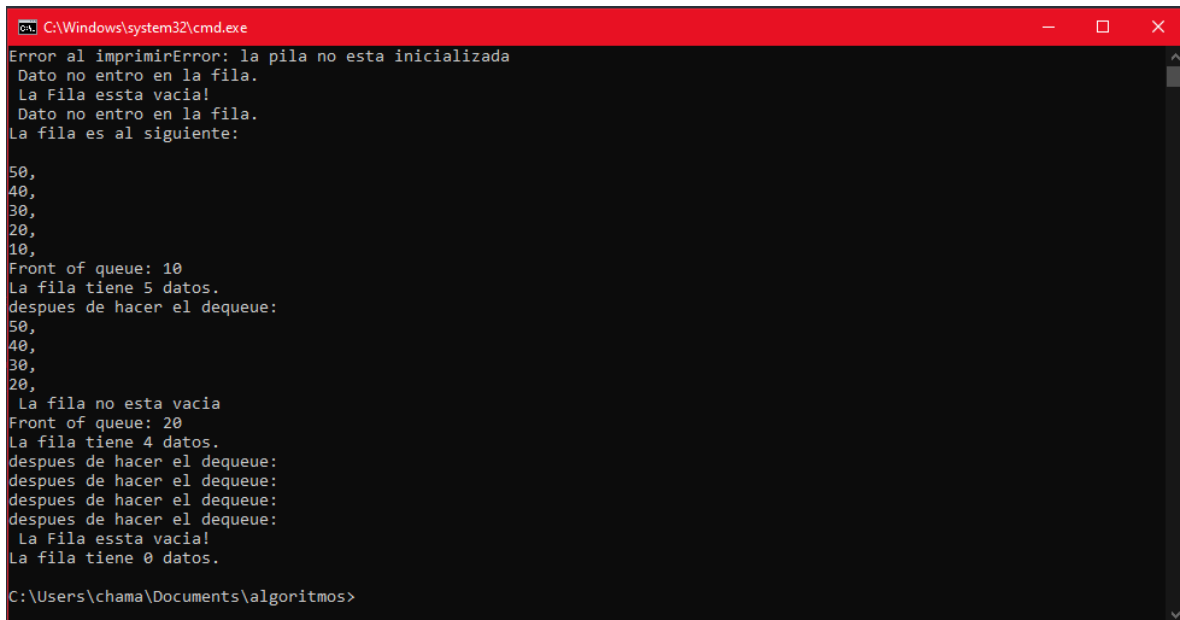
200         pf->Id_End++;
201         Flag = 1;
202     }
203     //Se retorna Flag
204     return Flag;
205 }
206
207 //Se inicializa la funcion printfila
208 void PrintFila(FILA *pf)
209 {
210     //Se declaran las variables de esta funcion
211     int k;
212
213     //Si no se ha inicializado la fila
214     if(pf==NULL)
215     {
216         printf("Error al imprimir");
217         return;
218     }
219     //Imprime los datos de la fila
220     for(k=pf->Id_End-1;k>=(signed int)pf->Id_Front;k--)
221         printf("\n%i, ",pf->data[k]);
222 }
223
224 //Se inicializa la funcion crear fila del tipo FILA
225 FILA* CrearFila(unsigned int Tam)
226 {
227     //Se declaran las variables de la funcion
228     FILA *ptr;
229
230     //Se reserva memoria para la fila
231     ptr=(FILA *)malloc(sizeof(FILA));
232
233     //Error al reservar memoria
234     if(ptr==NULL)
235     {
236         printf("Error al reservar memoria para la fila. ");
237         exit(0);
238     }
239
240     //Se asignan valores a las funciones
241     ptr->Id_Front=0;
242     ptr->Id_End=0;
243     ptr->Size=Tam;
244     //Se reserva memoria
245     ptr->data=(int *)malloc(Tam*sizeof(int));
246
247     //Error al reservar memoria
248     if (ptr->data==NULL)
249     {

```

```
249     {
250         printf("Error al reservar memoria para los datos de la pila");
251         exit(0);
252     }
253     //Se retorna la variable ptr
254     return ptr;
255 }
256
257 //Se incializa la funcion freefila
258 void FreeFila(FILA *pf)
259 {
260     if(pf!=NULL)
261     {
262         //Se libera la fila
263         free(pf->data);
264         free(pf);
265     }
266     //Error al liberar la fila
267     else
268         printf("\n Error al liberar la fila");
269 }
```

Pruebas y resultados

Evidencia del programa



```
C:\Windows\system32\cmd.exe
Error al imprimirError: la pila no esta inicializada
Dato no entro en la fila.
La Fila essta vacia!
Dato no entro en la fila.
La fila es al siguiente:
50,
40,
30,
20,
10,
Front of queue: 10
La fila tiene 5 datos.
despues de hacer el dequeue:
50,
40,
30,
20,
La fila no esta vacia
Front of queue: 20
La fila tiene 4 datos.
despues de hacer el dequeue:
despues de hacer el dequeue:
despues de hacer el dequeue:
despues de hacer el dequeue:
La Fila essta vacia!
La fila tiene 0 datos.
C:\Users\chama\Documents\algoritmos>
```

Primero quise imprimir la pila, pero no me imprimió nada ya que no se había creado la pila aun, así que el dato que intenté ingresar no se pudo, entonces la pila está vacía, luego cree la fila y le ingresé 6 valores, pero como cree la fila con solo 5, no se ingresó el valor número 6 los cuales luego me imprimió, el primer dato es el número 10 y eso me lo imprime con la función frontofqueue y me marca la cantidad de datos que tiene la fila que son 5, aplicando la función dequeue, se va el 10 y se imprimen los demás valores, y ahora el front es el 20 y me lo imprime con la función frontofqueue y me imprime los 4 datos que son ahora y nos marca que la fila no está vacía, ya que aún tiene 4 datos, entonces aplicamos la función dequeue 4 veces mas para eliminar todos los datos y ahora con la función isempty nos marca que la fila está vacía y con la función size nos marca que ahora que se han eliminado todos los datos nos marca que tiene 0 datos la fila.