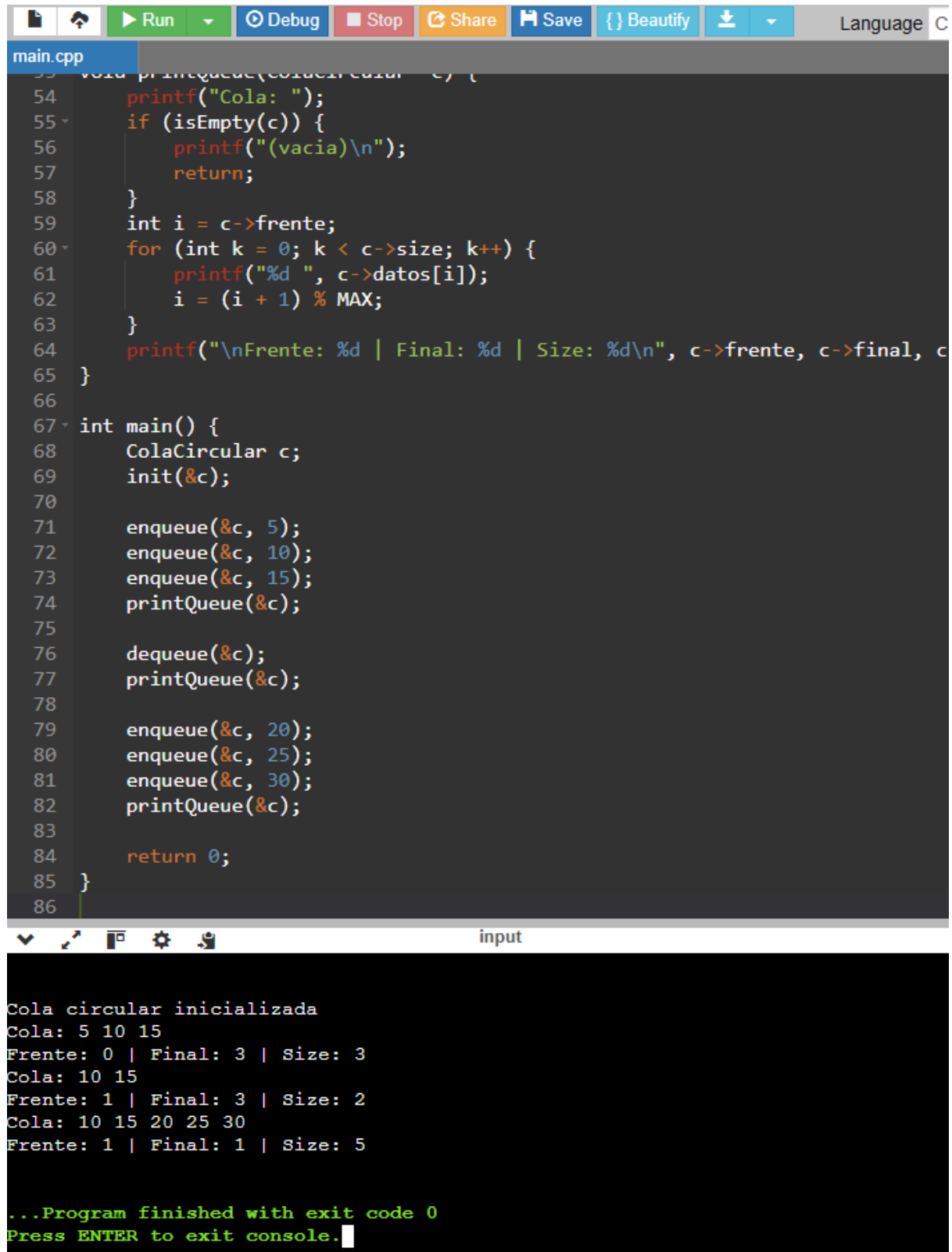


Ramirez Manriquez Luis Fernando

Capturas



The image shows a C++ IDE with a code editor and a console window. The code editor displays the implementation of a circular queue in C++.

```
main.cpp
53 void printQueue(ColaCircular c) {
54     printf("Cola: ");
55     if (isEmpty(c)) {
56         printf("(vacía)\n");
57         return;
58     }
59     int i = c->frente;
60     for (int k = 0; k < c->size; k++) {
61         printf("%d ", c->datos[i]);
62         i = (i + 1) % MAX;
63     }
64     printf("\nFrente: %d | Final: %d | Size: %d\n", c->frente, c->final, c->size);
65 }
66
67 int main() {
68     ColaCircular c;
69     init(&c);
70
71     enqueue(&c, 5);
72     enqueue(&c, 10);
73     enqueue(&c, 15);
74     printQueue(&c);
75
76     dequeue(&c);
77     printQueue(&c);
78
79     enqueue(&c, 20);
80     enqueue(&c, 25);
81     enqueue(&c, 30);
82     printQueue(&c);
83
84     return 0;
85 }
86
```

The console window shows the output of the program:

```
Cola circular inicializada
Cola: 5 10 15
Frente: 0 | Final: 3 | Size: 3
Cola: 10 15
Frente: 1 | Final: 3 | Size: 2
Cola: 10 15 20 25 30
Frente: 1 | Final: 1 | Size: 5
...Program finished with exit code 0
Press ENTER to exit console.
```

```
main.cpp
85         cout << "(vacio)" << endl;
86     } else {
87         int i = front;
88         for (int k = 0; k < size; k++) {
89             cout << datos[i] << " ";
90             i = (i + 1) % MAX;
91         }
92         cout << endl;
93     }
94
95     cout << "Front: " << front
96          << " | Rear: " << rearIndex()
97          << " | Size: " << size << endl;
98 }
99 };
100
101 int main() {
102     Deque d;
103
104     d.insert_front(10);
105     d.insert_rear(15);
106     d.insert_rear(20);
107     d.insert_front(30);
108     d.insert_rear(40);
109
110     d.printDeque();
111
112     d.delete_front();
113     d.delete_rear();
114
115     d.printDeque();
116
117     return 0;
118 }
119
input
Deque: 30 10 15 20 40
Front: 3 | Rear: 2 | Size: 5
Deque: 10 15 20
Front: 4 | Rear: 1 | Size: 3
...Program finished with exit code 0
Press ENTER to exit console.
```

Actividad

Preguntas (A)

1. ¿Por qué en la cola circular no se desperdician posiciones como en la cola lineal?

Porque la cola circular “da la vuelta” cuando llega al final. En vez de dejar huecos inutilizables, vuelve al inicio y sigue usando esas posiciones, así que el espacio se aprovecha al máximo.

2. ¿Qué pasa cuando el índice final llega al final del arreglo?

Literalmente solo “se reinicia”: si llega al último espacio, salta al índice 0. No se sale del arreglo, solo gira como un círculo.

3. ¿Qué sucede si intentas encolar cuando la cola está llena?

Pues simplemente no te deja. La estructura detecta que ya no cabe nada más y te marca que está llena, así que no mete el nuevo dato.

Preguntas (B)

1. ¿Qué diferencia tiene el deque frente a la cola circular normal?

El deque es más flexible porque puedes meter y sacar datos tanto por el frente como por atrás. La cola circular solo deja meter por atrás y sacar por el frente.

2. ¿Qué operaciones permiten simular tanto FIFO como LIFO?

Si usas el frente y el final estratégicamente, puedes comportarte como una cola normal (FIFO) o como una pila (LIFO). Todo depende de por dónde metas y saques los valores.

3. ¿Qué aplicaciones prácticas tiene deque?

Sirve mucho en cosas donde necesitas mover datos por los dos lados, como navegadores (adelante/atrás), simulaciones, buffers de tareas o algoritmos donde necesitas manejar elementos recientes y viejos rápidamente.

