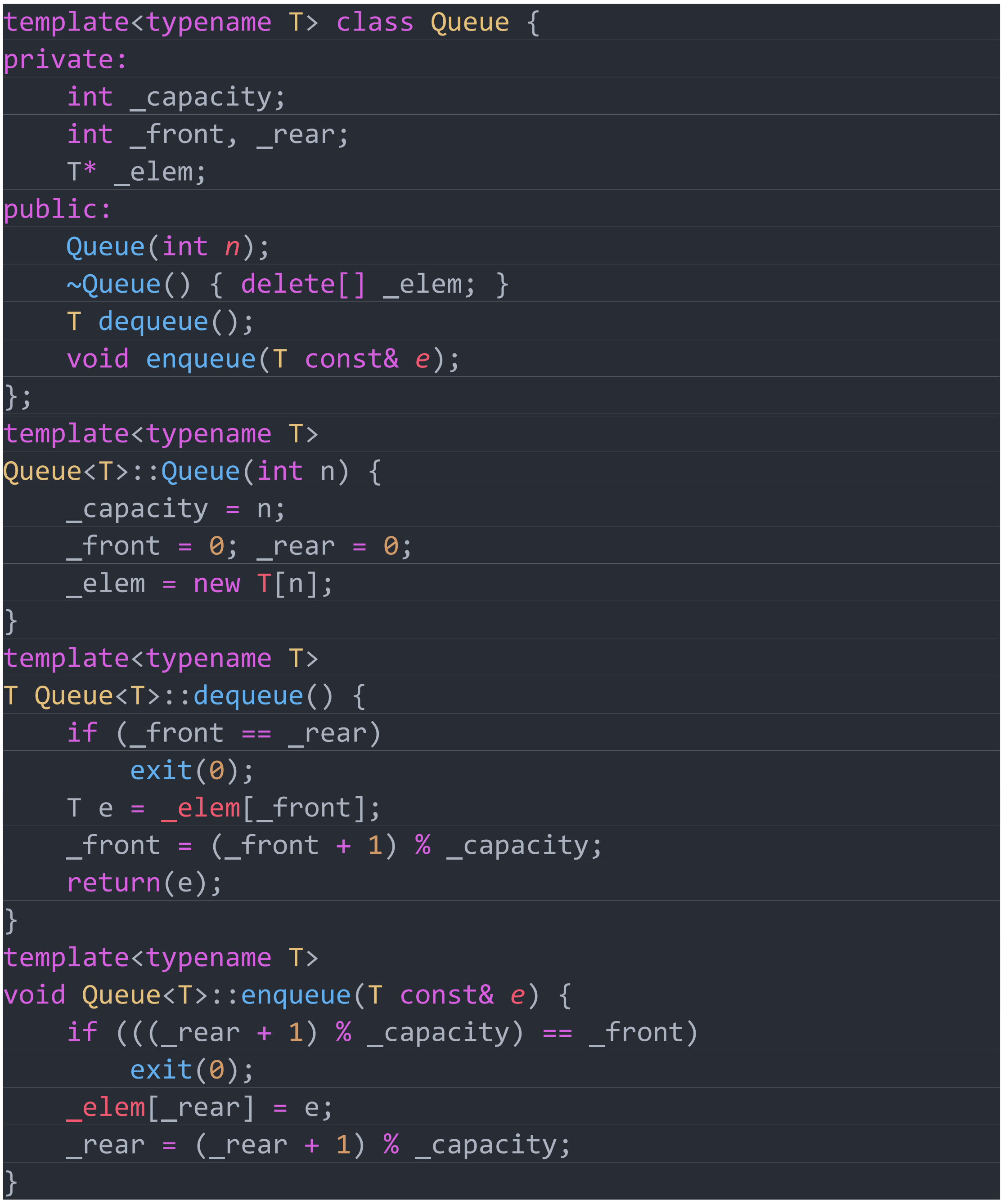

```
template<typename T> class Queue {
private:
    int _capacity;
    int _front, _rear;
    T* _elem;
public:
    Queue(int n);
    ~Queue() { delete[] _elem; }
    T dequeue();
    void enqueue(T const& e);
};
template<typename T>
Queue<T>::Queue(int n) {
    _capacity = n;
    _front = 0; _rear = 0;
    _elem = new T[n];
}
template<typename T>
T Queue<T>::dequeue() {
    if (_front == _rear)
        exit(0);
    T e = _elem[_front];
    _front = (_front + 1) % _capacity;
    return(e);
}
template<typename T>
void Queue<T>::enqueue(T const& e) {
    if (((_rear + 1) % _capacity) == _front)
        exit(0);
    _elem[_rear] = e;
    _rear = (_rear + 1) % _capacity;
}
```