

Single Queue:

```
#include <stdio.h>
#define MAX_SIZE 10
int queue[MAX_SIZE];
int front = 0;
int rear = -1;
int is_full() {
    return rear >= MAX_SIZE - 1;
}
int is_empty() {
    return front > rear;
}
void enqueue(int value) {
    if (is_full()) {
        printf("큐가 다 차버렸당께네~!\n");
        return;
    }
    queue[++rear] = value;
}
int dequeue() {
    if (is_empty()) {
        printf("아따 큐 비었자네 못혀\n");
        return -1;
    }
    return queue[front++];
}
int main() {
    enqueue(10);
    enqueue(20);
    enqueue(30);
    printf("뺀 값 %d\n", dequeue());
    printf("뺀 값 %d", dequeue());
    return 0;
}
```

Microsoft Visual S

뺀 값 10

뺀 값 20

C:\Users\user\D

: 0개).

이 창을 닫으려

Circular Queue:

```
        return (rear + 1) % MAX_SIZE == front;
    }

    int is_empty() {
        return front == rear;
    }

    void enqueue(int value) {
        if (is_full()) {
            printf("큐가 가득 찼습니다.\n");
            return;
        }

        queue[rear] = value;
        rear = (rear + 1) % MAX_SIZE;
    }

    int dequeue() {
        if (is_empty()) {
            printf("큐가 비어있습니다.\n");
            return -1;
        }

        int value = queue[front];
        front = (front + 1) % MAX_SIZE;
        return value;
    }

    int menu(void)
    {
        int n;
        printf("\n===== \n");
        printf("1. 큐에 값을 입력\n");
        printf("2. 큐를 출력하고 초기화\n");
        printf("3. 종료\n");
        printf("===== \n");
        scanf_s("%d", &n);
        return n;
    }

    int main() {
        int n;
```

C:\Users\Wuser\WDesktop\W202

```
=====
1. 큐에 값을 입력
2. 큐를 출력하고 초기화
3. 종료
=====
```

10

```
=====
1. 큐에 값을 입력
2. 큐를 출력하고 초기화
3. 종료
=====
```

1

값을 입력하세요: 10

```
=====
1. 큐에 값을 입력
2. 큐를 출력하고 초기화
3. 종료
=====
```

|