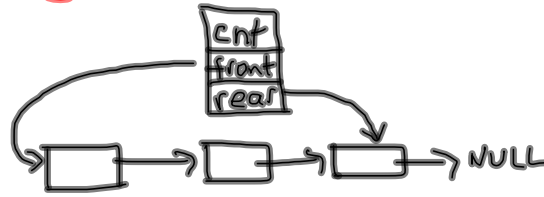


Kuyruklar (Queues)

FIFO (First in First out)

1) Bağlı Listeler ile Implementasyonu



```
typedef struct {
    int cnt;
    struct node *front;
    struct node *rear;
} queue;

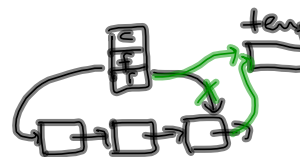
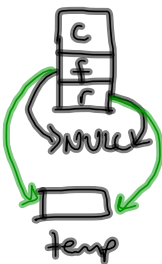
void initialize (queue *q) {
    q->cnt = 0;
    q->front = q->rear = NULL;
}

int isempty (queue *q) {
    return (q->cnt == 0);
}
```

Kuyruk işlemleri

1) Enqueue (Kuyruğa Ekleme)

```
void enqueue (queue *q, int x) {
    if (!isfull (q)) {
        struct node *temp = new node();
        temp->next = NULL;
        temp->data = x;
        q->cnt++;
        if (isempty(q))
            q->front = q->rear = temp;
        else {
            q->rear->next = temp;
            q->rear = temp;
        }
    }
}
```

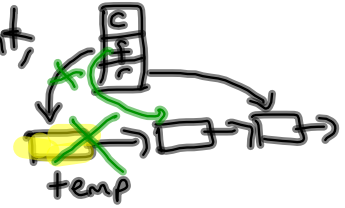


2-) Dequeue (Kuyruktan çıkarma)

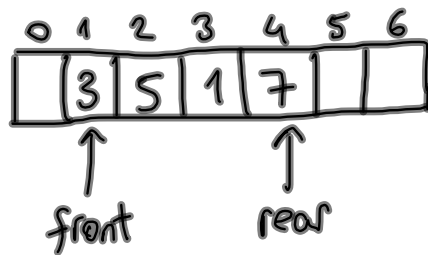
```

int dequeue (queue *q) {
    if (!isEmpty(q)) {
        struct node *temp = q->front;
        int x = temp->data;
        q->front = temp->next;
        free(temp);
        q->cnt--;
        return x;
    }
}

```



② Array ile Implementasyon



```

#define QUEUE_SIZE 10
typedef struct {

```

```

    int front;
    int rear;
    int cnt;
    int data[QUEUE_SIZE];

```

```

} queue;

```

```

void initialize (queue *q) {

```

```

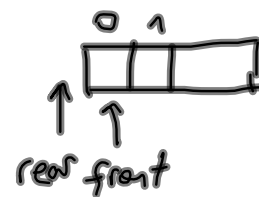
    q->front = 0;
    q->rear = -1;
    q->cnt = 0;

```

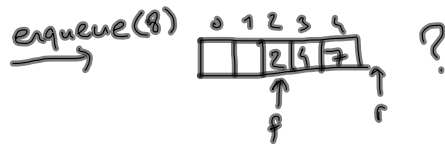
```

}

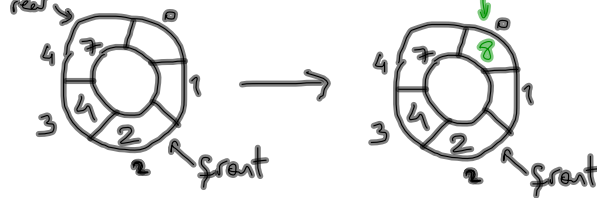
```



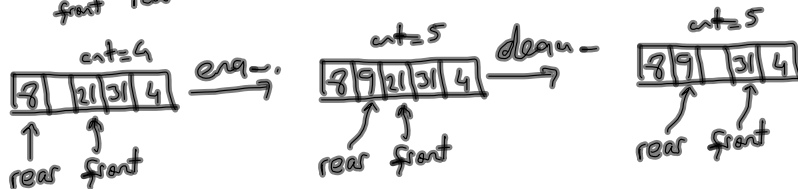
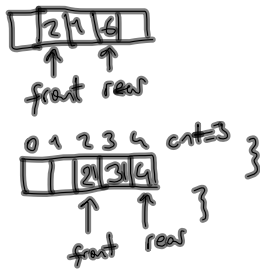
Array Imp. ile Kuyruk işlemleri



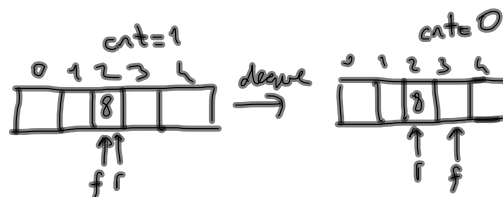
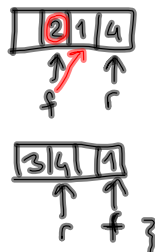
Çözüm: Circular Array



1.) void enqueue(queue *q, int x) {
 if (!isfull(q)) {
 q->rear++;
 q->cnt++;
 if (q->rear == QUEUE_SIZE)
 q->rear = 0;
 q->data[q->rear] = x;
 }



2.) int dequeue(queue *q) {
 if (!isempty(q)) {
 int x = q->data[q->front];
 q->front++;
 q->cnt--;
 if (q->front == QUEUE_SIZE)
 q->front = 0;
 return x;
 }



Soru: Verilen bir stringin palindrome olup olmadığını bir stack ve bir queue kullanarak belirleyen bir program yazınız.

Örn: Ey edip Adana'da pide ye.

Not. Noktalama işaretleri, büyük harfler ve boşluklar ihmal edilir.

```
main(){
    string the_string;
    queue q;
    stack s;
    initialize_queue(&q);
    initialize_stack(&s);
    int i=0, mismatches=0;
    cout << "Bir string girin";
    cin >> the_string;
    while (i < the_string.length()){
        if (isalpha(the_string[i])){
            enqueue(&q, tolower(the_string[i]));
            push(&s, tolower(the_string[i]));
        }
        i++;
    }
    while (!isEmpty(q)){
        if (pop(&s) != dequeue(&q)){
            mismatches=1;
            break;
        }
    }
    if (mismatches==1)
        cout << "Palindrome değil";
    else
        cout << "Palindrome";
} //main için
```

