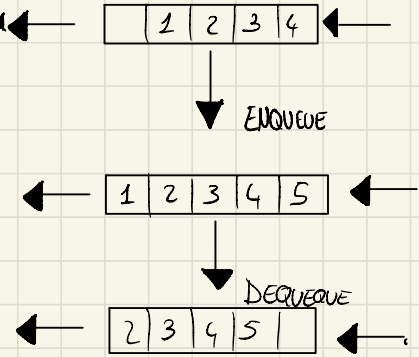


Coda:

Entero diverso dalle Pile, esse utilizzano il FIFO

First
In
First
Out



Diversi Metodi per implementarle (2)

Complessità Enqueue: $O(1)$

Complessità Dequeue: $O(1)$

La queue statica la possiamo pensare come un Array:

Dove abbiamo due variabili head e tail che ci indicano

dove togliere e inserire gli elementi:

Vediamo come strutturare il codice:

template <typename T>

class Queue_static

{

T* vet;

int head = 0;

int tail = -1;

int size = 0;

int max_size;

public:

Queue_static(int max_size): max_size(max_size) { vet = new T[max_size]; }

Veriamo i metodi:

```
bool isEmpty() {return size == 0;}  
bool isFull() {return size == max_size;}
```

```
void enqueue(T val)  
{
```

```
    if (isFull())  
    {
```

```
        cerr << "In Full Stack:" << endl;  
        return;
```

```
    }  
    tail = ++tail % max_size; (?)  
    vel[tail] = val; // inserimento del valore val  
    size++; // incremento il numero di valori presenti nella coda  
    return; // esco dal ciclo.  
}
```