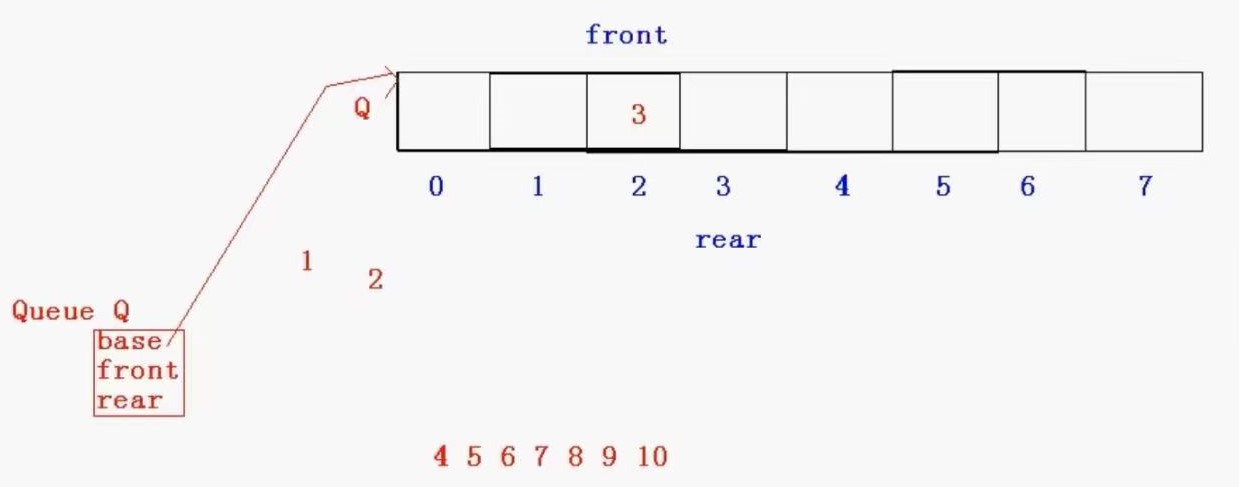
**队列之顺序队列的实现**

****

#include <stdio.h> //标准输入输出的头文件

#include <malloc.h>

#include <assert.h>

typedef int ElemType;

#define MAXSIZE 8

typedef struct Queue {

ElemType\* base;

int front;

int rear;

}Queue;

void InitQueue(Queue\* Q) {

Q->base = (ElemType\*)malloc(sizeof(ElemType) \* MAXSIZE);

assert(Q->base != NULL);

Q->front = Q->rear = 0;

}

void EnQueue(Queue\* Q,ElemType x) {

if (Q->rear >= MAXSIZE) { //入队判断队列满不满

return;

}

Q->base[Q->rear++] = x;

}

void ShowQueue(Queue\* Q) {

for (int i = Q->front; i < Q->rear; i++) {

printf("%d ", Q->base[i]);

}

printf("\n");

}

void DeQueue(Queue\* Q) {

if (Q->front == Q->rear) { //判断队列空不空的条件

return;

}

Q->front++;

}

void GetHead(Queue\* Q, ElemType\*v) {

if (Q->front == Q->rear) { //判断队列空不空的条件

return;

}

\*v = Q->base[Q->front];

}

int Length(Queue\* Q) {

return (Q->rear - Q->front);

}

void ClearQueue(Queue\* Q) {

Q->front = Q->rear = 0; //回归到初始设置

}

void DestroyQueue(Queue\* Q) {

ClearQueue(Q);

free(Q->base);

Q->base = NULL;

}

void main() {

Queue Q;

InitQueue(&Q);

for (int i = 1; i <= 5; i++) {

EnQueue(&Q, i);

}

ShowQueue(&Q);

DeQueue(&Q);

ShowQueue(&Q);

}