

数据结构

顺序栈的实现

✓ 数据结构 讲逻辑 存储 操作
觉其实分别对应 思想 空间 行动

创客学院 小美老师

顺序栈

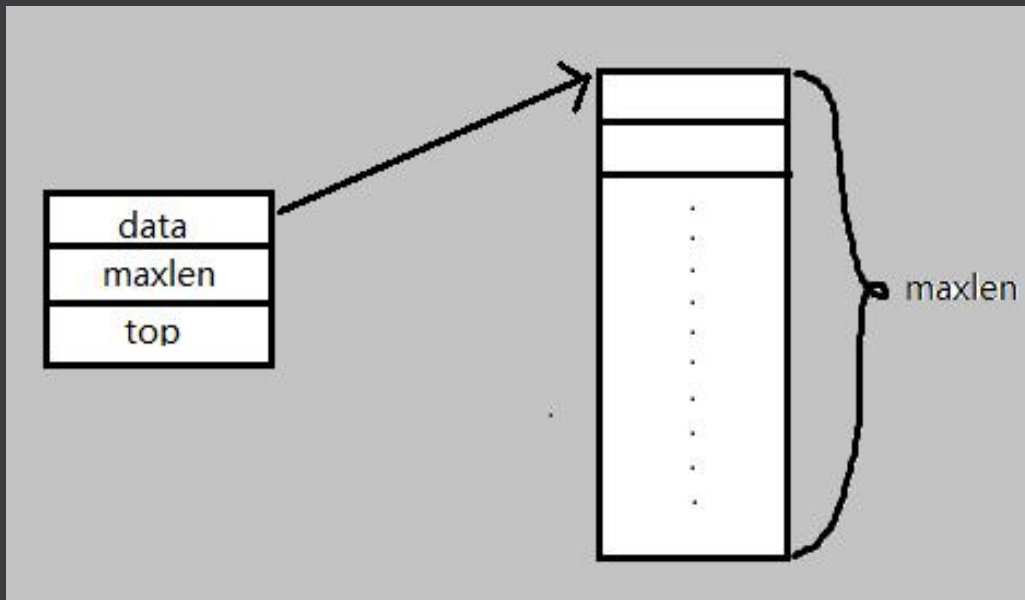
它是顺序表的一种，具有顺序表同样的存储结构，由数组定义，配合用数组下标表示的栈顶指针top（相对指针）完成各种操作。

```
typedef int data_t; /*定义栈中数据元素的数据类型*/
typedef struct
{
    data_t *data; /*用指针指向栈的存储空间*/
    int maxlen; /*当前栈的最大元素个数*/
    int top; /*指示栈顶位置(数组下标)的变量*/
} sqstack; /*顺序栈类型定义*/
```

栈

创建栈:

```
sqstack *stack_create (int len)
{
    sqstack *ss;
    ss = (sqstack *)malloc(sizeof(sqstack));
    ss->data = (data_t *)malloc(sizeof(data_t) * len);
    ss->top = -1;
    ss->maxlen = len;
    return ss;
}
```



栈

清空栈:

```
stack_clear(sqstack *s)
{
    s-> top = -1 ;
}
```

判断栈是否空 :

```
int stack_empty (sqstack *s)
{
    return (s->top == -1 ? 1 : 0);
}
```

栈

进栈：

```
void stack_push (sqstack *s , data_t x)
{
    if (s->top == N - 1) {
        printf ( "overflow !\n" );
        return ;
    }
    else {
        s->top ++ ;
        s->data[s->top] = x ;
    }
    return ;
}
```

栈

出栈：

```
datatype stack_pop(sqstack *s)
{
    s->top--;
    return (s->data[s->top+1]);
}
```

取栈顶元素：

```
datatype get_top(sqstack *s)
{
    return (s->data[s->top]);
}
```

扫一扫，获取更多信息



THANK YOU