数据结构第三章作业　2020101603　阮炜霖

3.3 stack

3.4 功能:(1)将栈 S 内元素倒置
(2)将栈中值为 e 的元素删掉，其他元素的顺序不变

3.12 char

3.13 将队列内元素倒置

3.15

```cpp
//阮炜霖 作业3.19
#include<bits/stdc++.h>
#define inf 0x3f3f3f3f
//#define int long long
using namespace std;
const int maxn=2e5+7;
const int mod=1e9+7;

//int read(){   int x=0,f=1;char ch=getchar();while(ch<'0'||ch>'9'){if(c


signed main(){
//  ios::sync_with_stdio(0);
//  cin.tie(0);cout.tie(0);
    string str;
    cin>>str;
    stack<char>stk;
    for(int i=0;i<str.length();i++){
        if(stk.empty()) stk.push(str[i]);
        else{
            if(str[i]==')'&&stk.top()=='(') stk.pop();
            else if(str[i]==']'&&stk.top()=='[') stk.pop();
            else if(str[i]=='}'&&stk.top()=='{') stk.pop();
            else stk.push(str[i]);
        }
    }
    if(stk.empty()) cout<<"配对正确\n";
    else cout<<"配对错误\n";
    return 0;
}

```

3.28

```
class node {
    public:
        node(int x):data(x) {
            nxt=NULL;
        }
        friend class Que;
    private:
        int data;
        node *lst,nxt;
};

class Que {
    public:
        inline Que() {
            tail=new node(0);
            tail->nxt=tail;
            tail->lst=tail;
        }
        inline ~Que() {
            node*p=tail;
            while(p!=tail) {
                node*q=p;
                delete q;
                p->nxt;
            }
            return;
        }
        inline void push(int data) {
            sz++;
            node*p=new node(data);
            node*q=tail->nxt;
            tail->nxt=p;
            p->lst=tail;
            p->nxt=q;
            q->lst=p;
            return;
        }
        inline void pop() {
            if(!sz) return;
            sz--;
            node*p=tail->lst;
            *q=tail->nxt;
            deleta tail;
            p->nxt=q;
            q->lst=p;
            tail=p;
            return;
        }
    private:
        node*tail;
        static int sz;
};
```