

# 字节跳动二面

## 自我介绍

## 实现一个小根堆

我想着实现一个指针版的，然后发现这个堆的最后一个元素这个有点不好维护，限于时间原因面试官中断了对那个的维护。

```

#include <iostream>
#include <algorithm>

struct node{
    node *p;
    node *l,*r;
    int value;
    node(){}
    node(node *p0,int value0):p(p0),l(nullptr),r(nullptr),value(value0){}
};

struct heap_info{
    node *root;
    node *last_parent;
}

node *goUpper(node *x) { // return the last item
    bool flag = true;
    node *last;
    if (x->p == nullptr || x->p->value <= x->value) {
        if (flag) return x; else return last;
    } else {
        swap(x->p->value,x->value);
        if (flag) {flag = false; return}
        x = x->p;
    }
}

void goDown(node *x) {
    if (x->l == nullptr || x->r == nullptr) return;
    if (x->r == nullptr || x->l->value <= x->r->value) {
        if (x->value > x->l->value) {
            swap(x->l->value,x->value);
            x = x->l;
        }
        // try go left
    } else {
        // rty go right
        if (x->value > x->r->value) {
            swap(x->r->value,x->value);
            x = x->r;
        }
    }
}

void addItem(heap_info *h, int value) {
    if (h->root == nullptr) {
        h->root = new node(nullptr,value);
        h->last_parent = h->root;
        return;
    }
}

```

```

node *newNode = new node(h->last_parent,value);
if (h->last_parent->l == nullptr) {
    h->last_parent->l = newNode;
} else {
    h->last_parent->r = newNode;
}

goUpper(newNode);
// todo:update h->last_parent
}

node* removeTop(heap_info *h) {
    node *last;
    if (h->last_parent->l == nullptr) {
        last = h->last_parent->l;
        h->last_parent->l = nullptr;
    } else {
        last = h->last_parent->r;
        h->last_parent->r = nullptr;
    }
    h->root->value = last->value;
    delete last;
    goDown(h->root);
    // todo:update h->last_parent
}

int main() {
}

```

**有一个栈和一个无序的整数数组，将这个数组按顺序入栈，要求在出栈的时候能输出栈内的最大元素值**

单调栈

尴尬，居然忘了stack的栈顶是top，一直打的是last()，后来突然像是top()。

然后填参数的时候，一直填的是a[i]编译出错一直没去看26行，还是面试官提醒。

```

#include <iostream>
#include <stack>
using namespace std;

stack<int> a,b;

void push(int x) {
    a.push(x);
    if (b.empty() || b.top() < x) {
        b.push(x);
    } else {
        b.push(b.top());
    }
}

void pop() {
    int realvalue = a.top(); a.pop();
    int maxvalue = b.top(); b.pop();
    cout<<realvalue<<" "<<maxvalue<<endl;
}

int arr[] = {1,10,3,15,2,6};

int main() {
    for (int i = 0; i < 6; ++i)
        push(arr[i]);
    for (int i = 0; i < 6; ++i)
        pop();
}

```