

暨南大学本科实验报告专用纸(附页)

堆 Heap

课程名称 数据结构 成绩评定
实验项目名称 堆 Heap 指导老师 干晓聪
实验项目编号 14 实验项目类型 设计性 实验地点 数学系机房
学生姓名 郭彦培 学号 2022101149
学院 信息科学技术学院 系 数学系 专业 信息管理与信息系统
实验时间 2024年6月13日上午 ~ 2024年7月13日中午

1. 实验目的

实现一个基础堆结构

2. 实验环境

计算机: PC X64

操作系统: Windows + Ubuntu20.0LTS

编程语言: C++: GCC std20

IDE: Visual Studio Code

3. 程序原理

堆是一个完全二叉树, 满足任意节点 a_i , 其子节点值均小于它, 这是小根堆。大根堆反之。

插入时, 在整棵树右下角插入, 并且按照大小逐个上浮。可以证明, 这样操作过后堆依然满足其性质。

删除时, 将根与右下角对换, 并按照左右大小下压根节点, 最后删除右下角节点即可。

可以证明两者复杂度均为 $O(\log_2 n)$

4. 程序代码

4.1. big_root_heap.h

```
1  #ifndef PRIORITY_QUEUE_HPP
2  #define PRIORITY_QUEUE_HPP
3
4  #include <vector>
5
6  #ifdef __PRIVATE_DEBUGGE
7  #include <iostream>
8  #endif
9
10 namespace myDS
11 {
12     template<typename VALUE_TYPE>
13     class big_root_heap{
14     private:
15         std::vector<VALUE_TYPE> h;
16
17         void flow(std::size_t x) {
18             while(x > 1 && h[x] > h[x/2]) {
19                 std::swap(h[x],h[x/2]);
20                 x >>= 1;
21             }
22         }
23
24         void drown(std::size_t x) {
25             while(x * 2 <= h.size()-1) {
26                 int t = x * 2;
27                 if(t + 1 <= h.size()-1 && h[t + 1] > h[t]) t ++;
28                 if(h[t] <= h[x]) break;
29                 std::swap(h[x],h[t]);
30                 x = t;
31             }
32         }
33     public:
34
35         big_root_heap() {h.push_back(0);}
36
37         ~big_root_heap() { }
38
39         void push(VALUE_TYPE t)
40         {
41             h.push_back(t);
42             flow(h.size()-1);
43         }
44
45         VALUE_TYPE top(){
46             return h[1];
47         }
48     }
```

暨南大学本科实验报告专用纸(附页)

```
49
50     VALUE_TYPE pop(){
51         auto t = this->top();
52         std::swap(h[1],h[h.size()-1]);
53         h.pop_back();
54         drown(1);
55         return t;
56     }
57
58 #ifdef __PRIVATE_DEBUG
59     void innerPrint(){
60         for(auto x:h) std::cout << x << " ";
61         std::cout << "\n";
62     }
63 #endif
64 };
65 } // namespace myDS
66
67
68 #endif
```

4.2. _PRIV_TEST.cpp

```
1  #include <iostream>
2  #define __PRIVATE_DEBUG
3  #include <Dev\14\big_root_heap.h>
4  using namespace std;
5
6  int main()
7  {
8      myDS::big_root_heap<int> piq;
9      while(1) {
10         string s;
11         cin >> s;
12         if(s == "push") {
13             int t;
14             cin >> t;
15             piq.push(t);
16         } else if(s == "pop") {
17             cout << piq.pop() << "\n";
18         } else if(s == "top") {
19             cout << piq.top() << "\n";
20         } else if(s == "p") {
21             piq.innerPrint();
22         }
23     }
24 }
```

5. 测试数据与运行结果

运行上述 `_PRIV_TEST.cpp` 测试代码中的正确性测试模块，得到以下内容：

```
push 5
push 4
push 1
push 3
push 2
p
0 5 4 1 3 2
pop
5
p
0 4 3 1 2
top
4
pop
4
pop
3
pop
2
pop
1
p
0
```

可以看出，代码运行结果与预期相符，可以认为代码正确性无误。