

柴祥 2030502202 计科 202

实验一 基本数据结构及递归复习

[发布人: 熊伟 截止时间: 2022-03-02 23:59:59]

实验一 基本数据结构及递归复习

目的:

1. 熟悉基本的数据结构, 如数组、栈、队列、链表等
2. 熟悉递归的应用

实验内容:

1. 约瑟环问题

圆桌上围坐着 $2n$ 个人。其中 n 个人是好人, 另外 n 个人是坏人。从第一个人开始数数, 数到第 m 个人, 立即赶走该人; 然后从被赶走的人之后开始数数, 再将数到的第 m 个人赶走.....依此方法不断赶走围坐在圆桌上的人。

预先应如何安排这些好人与坏人的座位, 能使得在赶走 n 个人之后, 圆桌上围坐的剩余的 n 个人全是好人?

2. 回文问题:

利用栈与队列, 判断一个句子是否回文。(如“madam, i am a man.”, 去掉标点便是回文)

3. 利用递归写一个汉诺塔的游戏。

4. 利用递归, 写出一个全排列, 如 $\{1, 2, 3\}$ 的全排列是 $\{1, 2, 3\}, \{1, 3, 2\}, \{2, 1, 3\}, \{2, 3, 1\}, \{3, 1, 2\}, \{3, 2, 1\}$

1. 约瑟环问题

```
#include <iostream>
#include "vector"
using namespace std;

int main(){
    int m,n,count=0;
    cin>>m>>n;
    cout<<"d="<<m<<"\t"<<n<<endl;
    vector<int> ring(2*n+1,0);

    int cc=1;
    for(int i=1;count<n;i++){
        if(cc%m==0&&ring[i%(2*n)]==0){
            ring[i%(2*n)]=1;
            count++;
            cc=0;
            cout<<i;
        }
        else if (ring[i%(2*n)]==4) {
            cc-=1;
        }
        cc++;
    }
```

```

cout<<endl;
for(int i=1;i<ring.size();i++){
    cout<<"["<<i<<"]"<<"\t";
}
cout<<endl;
for(int i=1;i<ring.size();i++){
    cout<<ring[i]<<"\t";
}
}

```

The screenshot shows a C++ IDE with a file named `JohnRing.cpp`. The code defines a `main` function that takes two integers `m` and `n` as input. It initializes a `vector<int>` named `ring` with size `2*n+1` and value `0`. It then iterates from `i=1` to `n`. For each `i`, it checks if `cc%m == 0` and `ring[i%(2*n)] == 0`. If true, it sets `ring[i%(2*n)] = 1` and increments `count`. If `count` reaches `m`, it prints `i` and resets `count` to `0`. If `ring[i%(2*n)] == 4`, it decrements `cc`. The output shows the input `3 5`, the value `d=3`, and the sequence `3691215`. Below this, a table shows the state of the `ring` array at each index from `[1]` to `[10]`.

```

1  #include <iostream>
2  #include "vector"
3  using namespace std;
4
5  int main(){
6      int m,n,count=0;
7      cin>>m>>n;
8      cout<<"d="<<m<<"\t"<<n<<endl;
9      vector<int> ring(2*n+1,0);
10
11     int cc=1;
12     for(int i=1;count<n;i++){
13         if(cc%m==0&&ring[i%(2*n)]==0){
14             ring[i%(2*n)]=1;
15             count++;
16             cc=0;
17             cout<<i;
18         }
19         else if (ring[i%(2*n)]==4) {
20             cc--;

```

PROBLEMS 23 OUTPUT DEBUG CONSOLE TERMINAL

```

3 5
d=3    5
3691215
[1]    [2]    [3]    [4]    [5]    [6]    [7]    [8]    [9]    [10]
0      1      1      0      1      1      0      0      1      0
PS C:\Users\97373\Desktop\code-works\leetcode>

```

2. 回文问题

```

class Solution {
public:
    bool isPalindrome(string s) {
        int l=0;
        int r=s.size()-1;

```

```



        while(l<r){

            while(l<r&&!legal(s,l))
                l++;
            while(l<r&&!legal(s,r))
                r--;
            if(l<r&&s[l]!=s[r]){
                cout <<l<<'t'<<r;
                return false;
            }

            l++;
            r--;
        }
        return true;
    }
    int legal(string& s,int i){
        if(s[i]>='0'&&s[i]<='9')
            return 1;
        else if(s[i]>='a'&&s[i]<='z'){
            return 1;
        }
        else if(s[i]>='A'&&s[i]<='Z'){
            cout<<"d="<<'s[i]';

            s[i]=tolower(s[i]);
            cout<<"d="<<'s[i]';
            return 2;
        }
        return 0;
    }
};

```

 力扣 学习 题库 讨论 竞赛 求职  商店

题目描述

评论 (1.4k)

题解 (2.1k)

提交记录

C++

智能模式

提交结果	执行用时	内存消耗	语言	提交时间	备注
通过	12 ms	7.1 MB	C++	2022/02/25 15:54	添加
通过	12 ms	7.1 MB	C++	2022/02/25 10:52	添加
执行出错	N/A	N/A	C++	2022/02/25 10:50	添加
执行出错	N/A	N/A	C++	2022/02/25 10:49	添加

```

1  class Solution {
2  public:
3      bool isPalindrome(string s) {
4          int l=0;
5          int r=s.size()-1;
6
7          while(l<r){
8              while(l<r&&!legal(s,l))
9                  l++;
10             while(l<r&&!legal(s,r))
11                 r--;
12             if(l<r&&s[l]!=s[r]){
13                 cout <<l<<'t'<<r;
14                 return false;
15             }
16         }
17     }

```

=====

3. 汉诺塔

```

class Solution {
public:
    void hanota(vector<int>& A, vector<int>& B, vector<int>& C) {
        movee(A,B,C,A.size());
    }
    void movee(vector<int>& A, vector<int>& B, vector<int>& C,int deep){

```

```

        if (deep<=1) {
            C.push_back(A[A.size()-1]);
            A.pop_back();
            return;
        }

        movee(A, C, B, deep-1);
        movee(A, B, C, 1);
        movee(B, A, C, deep-1);
    }
};

```

力扣 学习 题库 讨论 竞赛 求职 商店

Plus 会员 我是面

题目描述 评论 (270) 题解 (296) 提交记录

提交结果	执行用时	内存消耗	语言	提交时间	备注
通过	0 ms	6.7 MB	C++	2022/02/25 15:33	添加
通过	0 ms	6.4 MB	C++	2022/02/25 10:55	添加

```

1 class Solution {
2 public:
3     void hanota(vector<int>& A, vector<int>& B, vector<int>& C) {
4         movee(A,B,C,A.size());
5     }
6     void movee(vector<int>& A, vector<int>& B, vector<int>& C,int deep){
7         if (deep<=1) {
8             C.push_back(A[A.size()-1]);
9             A.pop_back();
10            return;
11        }
12
13        movee(A, C, B, deep-1);
14        movee(A, B, C, 1);
15        movee(B, A, C, deep-1);
16    }
17 };

```

=====

4.全排列

```

class Solution {
public:
    vector<vector<int>> permute(vector<int>& nums) {
        vector<vector<int>> result;
        if(nums.size()==1)
            return {nums};
        int t=nums.back();
        nums.pop_back();
        result=permute(nums);

        int ss=result.size();
        for(int i=0;i<ss;i++){
            vector<int> copyy(result[i].begin(),result[i].end());
            result.erase(result.begin());

            int ss2=copyy.size();
            for(int j=0;j<=ss2;j++){
                vector<int> copyy1(copyy.begin(),copyy.end());
                copyy1.insert(copyy1.begin()+j,t);
                result.push_back(copyy1);
            }
        }
        return result;
    }
};

```

题目描述	评论 (1.7k)	题解 (3.5k)	提交记录
提交结果	执行用时	内存消耗	语言
通过	8 ms	8 MB	C++
超出时间限制	N/A	N/A	C++

```
1 class Solution {
2 public:
3     vector<vector<int>> permute(vector<int>& nums) {
4         vector<vector<int>> result;
5         if(nums.size()==1)
6             return {nums};
7         int t=nums.back();
8         nums.pop_back();
9         result=permute(nums);
10
11         int ss=result.size();
12         for(int i=0;i<ss;i++){
13             vector<int> copyy(result[i].begin(),result[i].end());
14             result.erase(result.begin());
15
16             int ss2=copyy.size();
17             for(int j=0;j<=ss2;j++){
18                 vector<int> copyy1(copyy.begin(),copyy.end());
19                 copyy1.insert(copyy1.begin()+j,t);
20                 result.push_back(copyy1);
21             }
22         }
23         return result;
24     }
25 };
26
```

