数据结构实验报告 1

批注 [鑫李1]: 替换具体数字

递归

批注 [鑫李2]: 替换具体实验名

姓　　名：南亚宏

学　　号：2311788

目录

2025 年 10 月 22 日

一、 题目一

批注 [鑫李3]: 后续实验报告修改题目以其描述

1.1 题目表述

编写递归函数计算 Fibonacci 数列，能避免重复计算
输入：input.txt，仅包含一个整数 n（0－90）
输出：程序应能检查输入合法性，若有错误，输出错误提示“WRONG”；否则输出 F(n)。两种情况都输出一个回车（形成一个空行）。所有实例均应在 30 秒内输出结果。
提示：可用一数组保存 Fibonacci 数列，用一个特殊值表示还未计算出 Fibonacci 数，递归调用前先检查数组，若已计算，直接取用，不进行递归调用；若未计算，调用递归函数，计算完成后保存入数组。实际上，这种方法得到了 F(1)－F(n)，而不仅是 F(n)。

1.2 代码 1 的测试用例:

输入：0
输出：0
输入：90
输出：2880067194370816120
输入：-1
输出：WRONG

1.3 思路:

主要用到记忆化递归的方法。为避免重复计算，用一个数组 fib 来保存已经计算过的 Fibonacci 数。这样，每次计算 F(n)时，先检查 fib[n]是否已经计算过，如果已经计算过，直接返回 fib[n]；如果没有计算过，递归计算 F(n-1)和 F(n-2)，并将结果保存到 fib[n]中。

1.4 代码:

批注 [鑫李4]: 粘贴代码

```
#include <iostream>
#include <fstream>
#include <vector>
#include <string>
using namespace std;

// 定义一个数组保存 Fibonacci 数
```

```cpp
const int MAX_N = 90;
vector<long long> fib(MAX_N + 1, -1);

// 递归函数
long long fibonacci(int n) {
    if (n == 0) return 0;
    if (n == 1) return 1;
    if (fib[n] != -1) return fib[n];
    fib[n] = fibonacci(n - 1) + fibonacci(n - 2);
    return fib[n];
}

int main() {
    ifstream inputFile("input.txt");
    if (!inputFile.is_open()) {
        cerr << "Error opening input file." << endl;
        return 1;
    }

    int n;
    if (!(inputFile >> n) || n < 0 || n > 90) {
        cout << "WRONG" << endl;
    } else {
        cout << fibonacci(n) << endl;
    }
    cout << endl; // 输出一个空行
    inputFile.close();
    return 0;
}
```

1.5 测试用例截图

批注 [鑫李5]: 粘贴测试用例的截图

input.txt

```
1    0
```

问题　输出　调试控制台　终端　端口

```
PS C:\Users\Macro\Desktop\hw1> ./main.exe
0
```

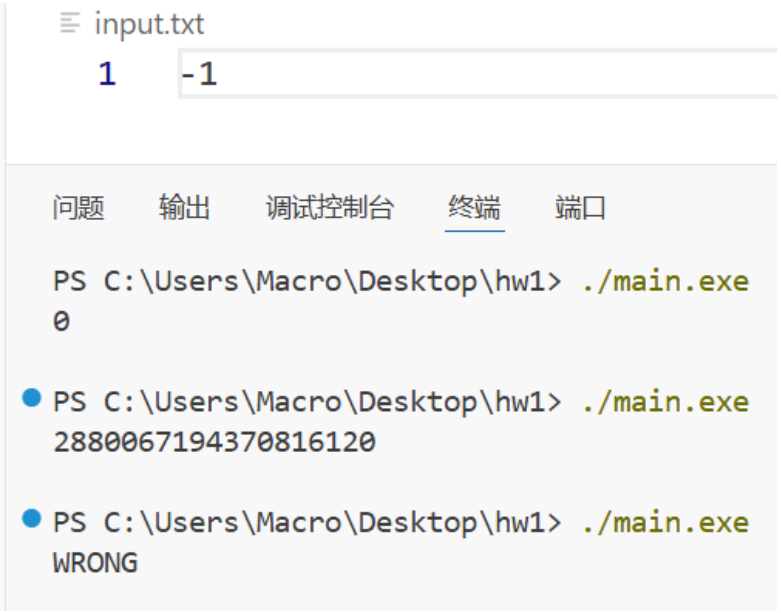
input.txt

```
1    90
```

问题　输出　调试控制台　终端　端口

```
PS C:\Users\Macro\Desktop\hw1> ./main.exe
0

PS C:\Users\Macro\Desktop\hw1> ./main.exe
2880067194370816120
```



二、 题目二

2.1 题目表述

编写递归函数，求 n 个元素集合的所有子集。不妨令集合元素为小写字母，原集合为{‘a’, ‘b’, …, ‘a’ + n - 1}。
输入：input.txt，仅包含整数 n（1－26）。
输出：若输入合法，输出集合的所有子集；否则输出“WRONG”。子集输出格式为每行一个子集，空集用空行表示，非空集合每个元素间用一个空格间隔，最后一个元素之后不能有空格。例如，对 n=3，可能的输出为：

2.2 代码 2 的测试用例:

输入：3
输出：a
a b
a b c
a c
b

b c
c
输入：27
输出：WRONG
输入：0
输出：WRONG

2.3 思路:

使用一个布尔数组来表示每个元素是否在子集中，状态为 true 表示包含该元素，状态为 false 表示不包含该元素。递归函数逐个考虑每个元素是否包含在子集中，当索引达到 n 时，表示已经考虑了所有元素，输出子集。

2.4 代码:

```
#include <iostream>
#include <fstream>
#include <vector>
#include <string>
using namespace std;

// 递归函数生成所有子集
void generateSubsets(int index, int n, vector<bool>& subset,
vector<char>& elements) {
    if (index == n) {
        // 输出子集
        bool isEmpty = true;
        for (int i = 0; i < n; ++i) {
            if (subset[i]) {
                cout << elements[i];
                isEmpty = false;
            }
            if (i < n - 1 && subset[i]) {
                cout << " ";
            }
        }
        if (isEmpty) {
            cout << endl; // 空集输出一行空行
        } else {
```

```
            cout << endl;
        }
        return;
    }
    subset[index] = false;
    generateSubsets(index + 1, n, subset, elements);

    subset[index] = true;
    generateSubsets(index + 1, n, subset, elements);
}

int main() {
    ifstream inputFile("input.txt");
    if (!inputFile.is_open()) {
        cerr << "Error opening input file." << endl;
        return 1;
    }

    int n;
    if (!(inputFile >> n) || n < 1 || n > 26) {
        cout << "WRONG" << endl;
    } else {
        vector<char> elements(n);
        for (int i = 0; i < n; ++i) {
            elements[i] = 'a' + i;
        }
        vector<bool> subset(n, false);
        generateSubsets(0, n, subset, elements);
    }
    inputFile.close();
    return 0;
}
```

2.5 测试用例截图

hw1.2 > input.txt

```
1 3
```

问题 输出 调试控制台 终端 端口

```
PS C:\Users\Macro\Desktop\hw1\hw1.2> ./main.exe

c
b
b c
a
a c
a b
a b c
PS C:\Users\Macro\Desktop\hw1\hw1.2>
```

hw1.2 > input.txt

```
1 27
```

问题 输出 调试控制台 终端 端口

```
PS C:\Users\Macro\Desktop\hw1\hw1.2> ./main.exe
b
b c
a
a c
a b
a b c
PS C:\Users\Macro\Desktop\hw1\hw1.2> ./main.exe
WRONG
```

hw1.2 > input.txt

```
1  0
```

问题 输出 调试控制台 终端 端口

```
PS C:\Users\Macro\Desktop\hw1\hw1.2> ./main.exe
b
b c
a
a c
a b
a b c
PS C:\Users\Macro\Desktop\hw1\hw1.2> ./main.exe
WRONG
PS C:\Users\Macro\Desktop\hw1\hw1.2> ./main.exe
WRONG
```