

資料結構 hw1

許家瑋

2024/10/18

1. 利用遞回阿克曼函數解決此問題及計算幕集的遞迴函數 實作完整程式碼放在.cpp 檔 此部分只放兩小題部分程式碼

1-1

```
3  ✓  int A(int m, int n) {  
4      if (m == 0) return (n + 1);  
5      if (n == 0) return A(m - 1, 1); //»%°j A(m - 1, 1)  
6      return A(m - 1, A(m, n - 1)); //»%°j A(m - 1, A(m, n - 1))  
7  }
```

1-2

```

3  ✓ int A(int m, int n) {
4      int a[1000] = { 0 }, b[1000] = { 0 };
5      int Top_a = 0, Top_b = 0;
6      a[Top_a++] = m;
7      b[Top_b++] = n;
8
9      while (Top_a != 0) {
10         if (m == 0) { // m == 0
11             n = n + 1;
12             while ((Top_a != 0) && (b[Top_b - 1] != -1)) {
13                 a[Top_a--] = 0;
14                 b[Top_b--] = 0;
15             }
16             if (Top_a != 0) {
17                 m = a[Top_a - 1];
18                 b[Top_b - 1] = n;
19             }
20         }
21         while (m != 0) { // m > 0
22             if (n == 0) { // n == 0
23                 m--;
24                 n = 1;
25                 a[Top_a++] = m;
26                 b[Top_b++] = n;
27             }
28             else { // n > 0
29                 n--;
30                 a[Top_a++] = m - 1;
31                 b[Top_b++] = -1;
32             }
33         }
34     }
35     return n;
36 }

```

2. 主程式部分

1-1

```

8  ✓ int main() {
9      int m, n;
10     cout << "Enter m and n: ";
11     cin >> m >> n;
12     cout << "A( " << m << " , " << n << " ) = " << A(m, n) << endl;
13     system("pause");
14     return 0;
15 }

```

1-2

```
37  ✓  int main() {  
38      int m, n;  
39      cout << "Enter m and n : ";  
40      cin >> m >> n;  
41      cout << "A( " << m << " , " << n << " ) = " << A(m, n) << endl;  
42      system("pause");  
43      return 0;  
44  }
```

3.

1-1

時間複雜度

$A(m,n)$ 當 m 越大為迭代指數級 難以表示

空間複雜度

遞迴深度隨著 n 的增長以指數級或更快的速度增加 導致空間複雜度也迅速上升 達到指數級甚至雙指數級的空間需求

1-2

時間複雜度

$O(n \cdot 2^n)$

空間複雜度

$O(n \cdot 2^n)$

因為幕集大小為指數級 因此時間空間的複雜度接受到幕集大小所影響

4.

心得

這是我第二次修江老師的資料結構 作業與之前大同小異 目前不算太難 反而是在實作時第一題有點卡住 所幸認真思考過後還是有成功編譯 唯一較為不同的是 這次要把檔案放在 **GitHub** 上 目前在作業上並無太大問題還算簡單