Homework 3

1. **Lập và giải công thức đệ quy xác định độ phức tạp thuật toán.**

1.

a)

x(n) = 2x(n − 3) for n > 1, x(1) = 1

= 22x(n – 6)

= 23x(n – 9)

= ...

= 2(n – 1)/3 x(n – 3((n - 1) /3))

= 2(n – 1)/3(1) = 2(n – 1)/3

=> x(n) = O(2n /3)

b)

x(n) = x(n − 2) − 2 for n > 1, x(1) = 0

= x(n – 4) – 2 – 2

= x(n – 6) – 2 – 2 -2

= ...

= x( n – 2(n - 1 /2)) – 2 \* (n - 1 /2)

= x(1) – n = 0 - n

=> x(n) = O(n)

2.

2.4.2

Setup: F(n) = F(n /2) + 1 if n > 1, F(0) = 1, F(1) = 1

a = 1, b = 2, d = 0 => a = bd

=> F(n) = O(logn)

1. **Lập trình đệ quy** *(Trong file đính kèm)*
2. **Đặt bài toán, thiết kế, phân tích và triển khai thuật toán**

Bài toán:

Bài toán chia thưởng: Có m phần thưởng được thưởng cho n học sinh giỏi có xếp

hạng theo thứ tự từ 1 đến n. Hỏi có bao nhiêu cách chia các phần thưởng thoả mãn các

điều kiện sau: (i) Học sinh giỏi hơn có số phần thưởng không ít hơn bạn kém hơn; (ii) m

phần thưởng phải chia hết cho các học sinh.

* **Phân tích bài toán**:
  + Nếu m = 0, hoặc n = 1, số cách chia là 1.
  + Nếu m < n, số cách chia bằng số cách chia m phần thưởng cho nngười.
  + Nếu m >= n, số cách chia bằng tổng của 2 trường hợp:
    - Chia i phần thưởng cho j – 1 người.
    - Chia i – j phần thưởng cho j người.
* **Xây dựng thuật toán:**
  + T(m, n) =
* **Chứng minh tính đúng của thuật toán**
* Trường hợp cơ sở:

Khi m = 0 hoặc n = 1, số cách chia phần thưởng là 1, tức là T(m,n) = 1. Điều này đúng vì khi không có phần thưởng để chia (m = 0) hoặc chỉ có 1 người (n = 1), không có cách nào để chia phần thưởng.

* Trường hợp giả sử:

Giả sử định lý đúng với giá trị m' < m và n' < n, tức là T(m', n') được tính đúng theo công thức đã cho.

* Trường hợp quy nạp:
* Khi m >= n, số cách chia phần thưởng được chia thành 2 trường hợp:
* Trường hợp 1: Người cuối cùng không nhận phần thưởng. Khi đó, ta phải chia m phần thưởng cho n - 1 người còn lại, tức là T(m, n-1).
* Trường hợp 2: Người cuối cùng nhận phần thưởng. Khi đó, ta phải chia m - n phần thưởng cho n người còn lại, tức là T(m-n, n). Do đó, số cách chia phần thưởng là T(m, n) = T(m,n-1) + T(m-n,n).
* Khi m < n, tất cả các người đều nhận được phần thưởng, và mỗi người nhận được một phần thưởng. Do đó, số cách chia phần thưởng là T(m, m).

**=> Vậy ta đã chứng minh tính đúng của thuật toán bằng phương pháp quy nạp.**

* **Độ phức tạp của thuật toán**
* T(m, n) = T(m, n -1) + T(m – n, n) *(m >= n)*
* T(m, m) = T(m, m -1) + T(0, m) (m < n)

T(m, n) = T(m, n - 2) + T(m - n, n - 1) + T(m - 2n, n) *(m >= 2n)*

= T(m, n - 3) + T(m - n, n - 2) + T(m - 2n, n - 1) + T(m - 3n, n) *(m >= 3n)*

= ...

= T(m, 1) + T(m - n, 2) + ... + T(m - (m - n)n, m - n) + T(m - (m – n)n, m - n+1)

*(m >= (m – n)n)*

= O(mn)

T(m, m) = T(m, m-1) + T(0, m)

= T(m, m-2) + T(m-m, m-1) + T(0, m-1) + T(0, m)

= T(m, m-3) + T(m-m, m-2) + T(0, m-2) + T(m-2m, m-1) + T(0, m-1) + T(0, m)

= ...

= T(m, 1) + T(m-m, 2) + ... + T(m-(m -1)m, m -1) + T(0, m – 1 +1) + ... + T(0, m-1) + T(0, m)

= O(m)

**=> Độ phức tạp của thuật toán là O(mn)**

* **Chương trình mô tả** *(trong file đính kèm)*