

Phân tích & Thiết kế thuật toán (Algorithms Design & Analysis)

L/O/G/O

GV: HUỖNH THỊ THANH THƯỜNG

Email: thuonghtt@uit.edu.vn

TỔNG QUAN

CHƯƠNG 1



L/O/G/O

www.themegallery.com

Bài tập về nhà HW#01

MỘT SỐ LƯU Ý

- ❖ Bài tập đầu tiên này nhằm giúp SV “làm quen lại” với cách thức làm toán, trình bày và lập luận logic trong toán học.
- ❖ Trình bày bài làm như cách cô đã hướng dẫn trên lớp, phải có lập luận toán học chứ không phải chỉ đưa ra đáp án. Phải giải thích tại sao đếm ra được con số này...
- ❖ Cách thức nộp bài thế nào thì xem tại Assignment trên course.
- ❖ **LÀM ĐƯỢC BAO NHIÊU THÌ LÀM, KHÔNG BẮT BUỘC SV LÀM HẾT**

Nhắc lại bài học tuần 1&2

- ❖ Mathematical analysis = Count the number of times “each of operations” or “basic operation” is executed.
- ❖ (Dịch: Đánh giá tính hiệu quả về mặt thời gian của thuật toán bằng phương pháp toán học = Đếm tất cả số phép toán hoặc số phép toán cơ bản được thực hiện trong thuật toán đó)
- ❖ Basic operation = the most important operation, contribute the most to total running time
- ❖ (Dịch: Phép toán cơ bản là phép toán quan trọng nhất, “tích cực nhất”, tốn nhiều chi phí thực hiện nhất hoặc có thể hiểu là phép toán mà số lần thực hiện không ít hơn các phép toán khác)

Bài tập về nhà HW#01

- ❖ Ôn tập 1 số kỹ thuật toán sơ cấp:
 - Tính tổng hữu hạn
 - Phương pháp đếm

Cấp số cộng

❖ **Cấp số cộng**: dãy số thoả mãn điều kiện: 2 phần tử liên tiếp nhau sai khác nhau một hằng số (công sai)

- **Số hạng thứ n :**

$$a_n = a_1 + (n - 1)d$$

- **Tổng của n số hạng đầu:**

$$S_n = a_1 + a_2 + \cdots + a_n = \frac{n(a_1 + a_n)}{2}$$

(a_1 : phần tử đầu, d : công sai)

$$= \frac{n[2a_1 + (n - 1)d]}{2}$$

Cấp số nhân

❖ **Cấp số nhân**: dãy số thoả mãn điều kiện: tỷ số của hai phần tử liên tiếp là hằng số (công bội)

- **Số hạng thứ n:**

$$a_n = a r^{n-1}$$

- **Tổng các phần tử của cấp số nhân:**

$$S_{n+1} = \sum_{k=0}^n ar^k = \frac{a(1 - r^{n+1})}{1 - r}$$

(*a*: phần tử đầu, *r*: công bội)

$$\sum_{k=m}^n ar^k = \frac{a(r^m - r^{n+1})}{1 - r}$$

Tổng hữu hạn

❖ Tính chất:

Sum Manipulation Rules

1. $\sum_{i=l}^u ca_i = c \sum_{i=l}^u a_i$
2. $\sum_{i=l}^u (a_i \pm b_i) = \sum_{i=l}^u a_i \pm \sum_{i=l}^u b_i$
3. $\sum_{i=l}^u a_i = \sum_{i=l}^m a_i + \sum_{i=m+1}^u a_i$, where $l \leq m < u$
4. $\sum_{i=l}^u (a_i - a_{i-1}) = a_u - a_{l-1}$

Tổng hữu hạn

❖ Một số công thức cần nhớ:

Important Summation Formulas

$$1. \sum_{i=l}^u 1 = \underbrace{1 + 1 + \cdots + 1}_{u-l+1 \text{ times}} = u - l + 1 \text{ (} l, u \text{ are integer limits, } l \leq u \text{); } \sum_{i=1}^n 1 = n$$

$$2. \sum_{i=1}^n i = 1 + 2 + \cdots + n = \frac{n(n+1)}{2} \approx \frac{1}{2}n^2$$

$$3. \sum_{i=1}^n i^2 = 1^2 + 2^2 + \cdots + n^2 = \frac{n(n+1)(2n+1)}{6} \approx \frac{1}{3}n^3$$

$$4. \sum_{i=1}^n i^k = 1^k + 2^k + \cdots + n^k \approx \frac{1}{k+1}n^{k+1}$$

Tổng hữu hạn

❖ Một số công thức cần nhớ:

$$5. \sum_{i=0}^n a^i = 1 + a + \cdots + a^n = \frac{a^{n+1} - 1}{a - 1} \quad (a \neq 1); \quad \sum_{i=0}^n 2^i = 2^{n+1} - 1$$

$$6. \sum_{i=1}^n i2^i = 1 \cdot 2 + 2 \cdot 2^2 + \cdots + n2^n = (n - 1)2^{n+1} + 2$$

$$7. \sum_{i=1}^n \frac{1}{i} = 1 + \frac{1}{2} + \cdots + \frac{1}{n} \approx \ln n + \gamma, \text{ where } \gamma \approx 0.5772 \dots \text{ (Euler's constant)}$$

$$8. \sum_{i=1}^n \lg i \approx n \lg n$$

[1]Anany Levitin, Introduction to the Design and Analysis of Algorithms, 3rd Edition, 2014 (trang 476)

Cận trên, cận dưới

❖ Một số công thức cần nhớ:

Floor and Ceiling Formulas

floor of a real number x , denoted $\lfloor x \rfloor$, $\lfloor 3.8 \rfloor = 3$, $\lfloor -3.8 \rfloor = -4$, $\lfloor 3 \rfloor = 3$

ceiling of a real number x , denoted $\lceil x \rceil$, $\lceil 3.8 \rceil = 4$, $\lceil -3.8 \rceil = -3$, $\lceil 3 \rceil = 3$

1. $x - 1 < \lfloor x \rfloor \leq x \leq \lceil x \rceil < x + 1$
2. $\lfloor x + n \rfloor = \lfloor x \rfloor + n$ and $\lceil x + n \rceil = \lceil x \rceil + n$ for real x and integer n
3. $\lfloor n/2 \rfloor + \lceil n/2 \rceil = n$
4. $\lceil \lg(n + 1) \rceil = \lfloor \lg n \rfloor + 1$

Bài 1: Tính tổng hữu hạn

a. $1 + 3 + 5 + 7 + \cdots + 999$

b. $2 + 4 + 8 + 16 + \cdots + 1024$

c. $\sum_{i=3}^{n+1} 1$

d. $\sum_{i=3}^{n+1} i$

e. $\sum_{i=0}^{n-1} i(i+1)$

f. $\sum_{j=1}^n 3^{j+1}$

g. $\sum_{i=1}^n \sum_{j=1}^n ij$

h. $\sum_{i=1}^n 1/i(i+1)$

i. $\sum_{j \in \{2,3,5\}} (j^2 + j)$

j. $\sum_{i=1}^m \sum_{j=0}^n \sum_{k=0}^{100} (i+j)$

Đếm số phép gán và so sánh

Bài 2

```
s = 0;  
i = 1;  
while (i ≤ n) do  
    j = 1;  
    while (j ≤ i2) do  
        s = s + 1;  
        j = j + 1;  
    end do;  
    i = i + 1;  
end do;
```

Đếm số phép gán và so sánh

Bài 3

```
sum = 0
i = 1
while i ≤ n do
    j = n - i * i
    while j ≤ i * i do
        sum = sum + i * j
        j = j + 1
    endw
    i = i + 1
endw
```

P_i →

Đếm số phép gán và so sánh

Bài 4

```
float Alpha (float x, long n)
{
    long i= 1; float z = 0;
    while ( i ≤ n)
    {
        long j = 1; float t = 1;
        while (j ≤ i)
        {
            t = t*x;
            j = 2*j;
        }
        z = z+i*t;
        i=i+1;
    }
    return z;
}
```

Đếm số phép gán và so sánh

Bài 5

Đề thi cuối kỳ

```
sum = 0; i = 1;
while ( i ≤ n)
{
    j = n - i ;
    while (j ≤ 2* i )
    {
        sum = sum + i*j;
        j = j + 2;
    }
    k = i ;
    while ( k > 0)
    {
        sum = sum + 1;
        k = k / 2;
    }
    i = i + 1;
}
```


Đếm số phép gán và so sánh

Bài 6(*)

Đề thi cuối kỳ HKI 2018-2019 (CT Tài năng/CLC)

```
i = 1; count = 0;
while ( i ≤ 4n )
{
    x = (n - i)(i - 3n) ;
    y = i - 2n;
    j = 1;
    while ( j ≤ x )
    {
        count = count - 2;
        j = j + 2;
    }
    if (x > 0)
        if (y > 0)
            count = count + 1;
    i = i + 1;
}
```

Đếm số phép gán và so sánh

Bài 7(*)

```
i=1;  
count = 0;  
while(i<=4n)  
{  
    x=(n-i) (i-3n)  
    y=i-2n  
    j=1  
    while(j<=x)  
    {  
        if(i>=2y)  
            count = count -2  
        j=j+1  
    }  
    i=i+1  
}
```

Đếm số phép gán và so sánh

Bài 8(*)

Đề thi cuối kỳ (CT Tài năng/CLC)

```
i= 1;count = 0;
while (i ≤ 3*n)
{
    x = 2*n - i;
    y = i - n ;
    j = 1;
    while (j ≤ x)
    {
        if(j ≥ n)
            count = count - 1;
        j = j+1;
    }
    if (y > 0)
        if (x > 0)
            count = count + 1;
    i = i+1;
}
```

Đếm số phép gán và so sánh

Bài 9(*)

```
i=1;  
res=0;  
while i≤n do  
    j=1;  
    k=1;  
    while j ≤ i do  
        res=res+i*j;  
        k=k+2;  
        j=j+k;  
    endw  
    i=i+1;  
endw
```

P_i →

Đếm số phép gán và so sánh

Đề thi cuối kỳ (CT Tài năng/CLC) HỌC KỲ 2 – NĂM HỌC 2021-2022

Bài 10(*)

```
sum = 0; i=1; idx=-1;
while (i<=n)
{
    j=1;
    while(j<=n)
    {
        if( (i==j) && (i+j==n+1) )
            idx=i;
        sum=sum+a[i][j];
        j++;
    }
    i++;
}
if(idx !=-1)
    sum=sum-a[idx][idx];
```

BONUS Không bắt buộc

Yêu cầu chung cho 3 bài {11, 12, 13} sau:

- a) Đếm theo n số phép gán và số phép so sánh trong mỗi đoạn chương trình
- b) **Kiểm tra kết quả đếm bằng máy tính:** Không dùng các công thức kết quả đã đếm được ở câu a, **viết chương trình** để tính trực tiếp giá trị của các hàm Gán(n), So sánh(n) với $n = 1, 2, \dots, 20$. Sau đó thống kê theo dạng bảng để so sánh kết quả với công thức đã có nhờ câu a.

n	1	2	...	20
Gán(n)= ...(công thức đếm thủ công từ câu a)				
Gán (n) kết quả khi chạy chương trình				
So sánh (n)= ...(công thức đếm thủ công từ câu a)				
So sánh (n) kết quả khi chạy chương trình				

Đếm số phép gán và so sánh

Bài 11(**) BONUS Không bắt buộc

```
i = 1; ret = 0; s = 0;
while ( i ≤ n )
{
    j = 1;
    s = s + 1/i; // {số thực}
    while ( j ≤ s )
    {
        ret = ret + i*j;
        j = j + 1;
    }
    i = i + 1;
}
```

Đếm số phép gán và so sánh

Bài 12(**) **BONUS** Không bắt buộc

(Lưu ý:
hiện tại cô
chưa có
danh sách
nhóm nên
“số thứ tự
của nhóm”
có thể chọn
là 1 số bất
 kỳ trừ số 1)

```
i = 1; res = 0;  
while (i ≤ n) do  
    j = 1;  
    while (j ≤ i) do  
        res = res + i*j ;  
        j = j + 1;  
    end do;  
    i = i + số thứ tự của nhóm;  
end do;
```


Đếm số phép gán và so sánh

Bài 13(**) BONUS Không bắt buộc

```
sum := 0;  
i := n;  
while (i > 0) do  
    j := i;  
    while (j > 0) do  
        sum := sum + 1;  
        j := j - 1;  
    endw;  
    i = i div 2;  
endw;
```