3.7. Bài tập

Bài 3.1. Cho một danh sách *n* phần tử kiểu số nguyên có giá trị khác nhau. Hãy:

a. Xây dựng giải thuật xác định phần tử nhỏ nhất và phần tử lớn nhất.

b. Đánh giá độ phức tạp của giải thuật bởi số phép so sánh.

c. Đề xuất giải thuật hiệu quả hơn.

d. Đánh giá độ phức tạp của giải thuật trong câu c bởi số phép so sánh.

Bài 3.2. Giả sử đa thức *P(x)* bậc *n* được định nghĩa như sau:

$$P(x) = \sum_{i=0}^{n} a_i x^{n-i}.$$

a. Hãy xây dựng giải thuật đơn giản tính *P(x)*.

b. Đánh giá số phép cộng và phép nhân trong giải thuật trên.

c. Cải tiến giải thuật trong câu *a* và đánh giá giải thuật.

d. Xây dựng giải thuật tính đa thức theo lược đồ Hörner

$$P(x) = \sum_{i=0}^{n} a_i x^{n-i} = \left(\ldots\left(\left(a_0 x + a_1\right)x + a_2\right)x + \ldots a_{n-1}\right)x + a_n$$

và đánh giá giải thuật.

So sánh hiệu quả của các giải thuật khác nhau ở trên.

Bài 3.3. Hãy xác định độ phức tạp của giải thuật sắp xếp chèn dưới đây trong ba trường hợp: tốt nhất, xấu nhất và trung bình.

```
algorithm SapXepChen(a, n)
input: danh sách a gồm n số nguyên
output: danh sách a đã được sắp xếp
begin
   for j ← 2 to n do
      khoa ← a[j]
      i ← j - 1
   while (i > 0) and (a[i] > khoa) do
      a[i+1] ← a[i]
      i ← i - 1
   endwhile
      a[i+1] ← khoa
   endfor
end
```

Bài 3.4. Giải thuật A thực hiện $4n^2$ xử lý cơ bản và giải thuật B thực hiện $1000\log(n)$ xử lý cơ bản. Khi nào thì giải thuật B hiệu quả hơn giải thuật A ?

Bài 3.5. Chứng minh rằng:

a. $4n^2 + 7n + 1 = O(n^2)$.

b. $n^2 - 3n + 1 = \Omega(n)$.

c. $\log(2n + k) = \Theta(\log(n))$, với k là một hằng số.

d. $\lceil \log(n) \rceil = O(n)$.

e. $\log(n^k) = \Theta(\log(n))$, với k là một hằng số.

f. $\sum_{i=1}^{n} \log(i) = \Theta(n \log(n))$.

Bài 3.6. Chứng minh rằng với mọi $k \geq 1$, $a_k, a_{k-1}, \ldots, a_1, a_0 \in R$,

$$a_k n^k + a_{k-1} n^{k-1} + \ldots + a_1 n^1 + a_0 = O(n^k).$$

Bài 3.7. Giải các hệ thức truy hồi sau:

a. $a_n = 3a_{n-1} + 2$, với $a_1 = 1$.

b. $a_n = a_{n-1} + n - 1$, với $a_1 = 2$.

c. $a_n = 4a_{n-1} - 4a_{n-2}$, với $a_0 = 0$ và $a_1 = 1$.

d. $a_n = a_{n/2} + 3n + 2$, với $a_1 = 4$ và n là lũy thừa của 2.

e. $a_n = 4a_{n/3} + 2n - 1$, với $a_1 = 3$ và n là lũy thừa của 3.

f. $a_n = 3a_{n/2} + n^2 - n$, với $a_1 = 1$ và n là lũy thừa của 2.

Bài 3.8. Hãy đánh giá độ phức tạp của giải thuật tìm kiếm nhị phân dưới đây trong trường hợp xấu nhất.

```
algorithm TimKiemNhiPhan(a, x, l, r)
input: danh sách a[l,r] đã được sắp xếp tăng dần có
       chứa phần tử x
output: chỉ số của x trong danh sách a
begin
   if l = r then return (l)
```

```
    else
       m ← (l + r)/2
       if x ≤ a[m] then
           return (TimKiemNhiPhan(a, x, l, m))
       else
           return (TimKiemNhiPhan(a, x, m+1, r))
       endif
    endif
end
```

Bài 3.9. Hãy sắp xếp các hàm độ phức tạp dưới đây sao cho mỗi hàm là o nhỏ của hàm tiếp theo: $n^3\log n$, $(\log \log n)^3$, $n^{0.5}2^n$, $(n + 4)^{12}$.