

BÀI TẬP LÝ THUYẾT CẤU TRÚC DỮ LIỆU

Bài 1: Cho biết độ phức tạp thuật toán của các đoạn chương trình (viết bằng C) sau

```
1. int i, j;
   int tong = 0;
   for (i = 0; i < n; i++)
       tong += i;
   for (j = 0; j < n/2; j++)
       tong += j;
   printf("tong = %d", tong);
```

```
3. int i, j;
   int tong = 0;
   for (i = 0; i < n; i++)
       for (j = 0; j < 4; j++)
           tong += i*j;
   for (i = 0; i < m; i++)
       tong -= i;
   printf("tong = %d", tong);
```

```
5. int i = 0; int j,k;
   int tong = 0;
   do {
       for (j = 0; j < n; j++)
           tong += i*j;
       for (k = 0; k < m; k+=2)
           tong += k;
       i++;
   } while (i < n)
   printf("tong = %d", tong);
```

```
8. int i, j;
   int tong = 0;
   for (i = 0; i < n; i+=2)
       for (j = n; j > 0; j/=4)
           tong += i*j;
   for (i = 0; i < m+n; i++)
       tong -= i;
   printf("tong = %d", tong);
```

```
2. int i, j;
   int tong = 0;
   for (i = 0; i < n; i++)
       for (j = 0; j < n/2; j++)
           tong += i*j;
   printf("tong = %d", tong);
```

```
4. int i = 0; int j,k;
   int tong = 0;
   do {
       for (j = 0; j < n; j++)
           for (k = 0; k < m; k+=2)
               tong = tong + i*j + k;
       i++;
   } while (i < n)
   printf("tong = %d", tong);
```

```
7. int i = 0; int j,k;
   int tong = 0;
   do {
       for (j = 1; j < n; j*=2)
           tong += i*j;
       for (k = m; k > 0; k/=3)
           tong += k;
       i++;
   } while (i < n)
   printf("tong = %d", tong);
```

```
6. int i = 0; int j,k;
   int tong = 0;
   do {
       for (j = 1; j < 10; j*=2)
           tong += i*j;
       for (k = m; k > 0; k/=3)
           tong += k;
       i++;
   } while (i < 10)
   printf("tong = %d", tong);
```

Bài 2: Tính giá trị các biểu thức hậu tố sau:

1. $S = 1\ 2\ 3\ +\ +\ 5\ *\ 3\ 2\ -\ -$
2. $S = 4\ 2\ -\ 4\ *\ 2\ ^\ 8\ /\ 1\ 2\ 3\ +\ -\ +$
3. $S = 9\ 8\ 7\ -\ -\ 1\ 2\ 3\ *\ +\ 4\ 5\ 6\ -\ +\ -\ +$
4. $S = 1\ 2\ +\ 3\ 4\ *\ 6\ 7\ -\ 8\ 9\ *\ +\ *\ +$
5. $S = 9\ 8\ 7\ 6\ 5\ 4\ 3\ 2\ 1\ +\ -\ *\ +\ +\ -\ *\ +$

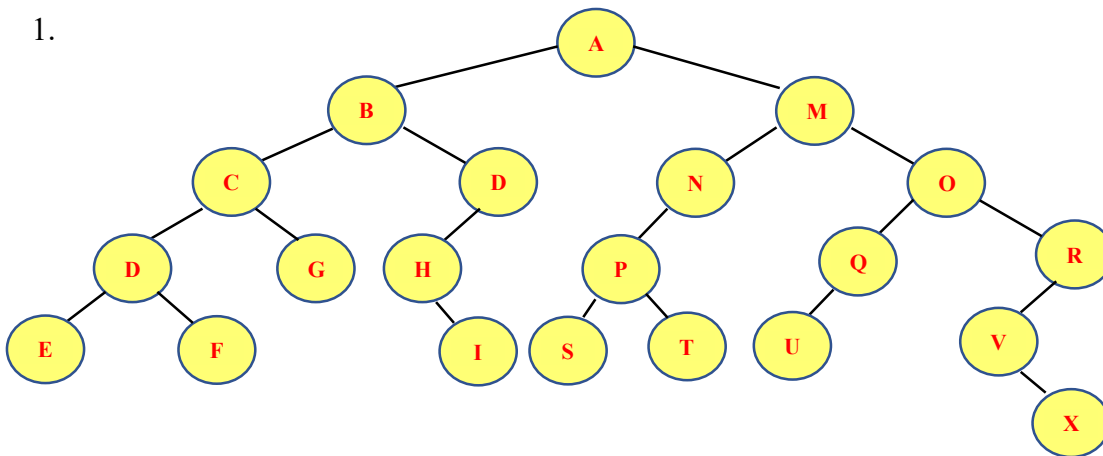
Bài 3: Chuyển đổi các biểu thức sau thành biểu thức hậu tố bằng phương pháp sử dụng Stack

1. $A/B + C*D*(E - F^G) - (H*I + K/L)^M$
2. $A + B*(C - D^E)/F + G^H - (I + J)*K^L$
3. $A*(B + C)^D - E^(F + G*H) + (I^J + K)/L$
4. $(A + (B*C - D)/E) + F*G/H - I/(K - L^(M + N))$
5. $A*(B - C*(D + E^F*(G^H - I) - K/L)) - (M*N + P/(R^S - T))^*V$

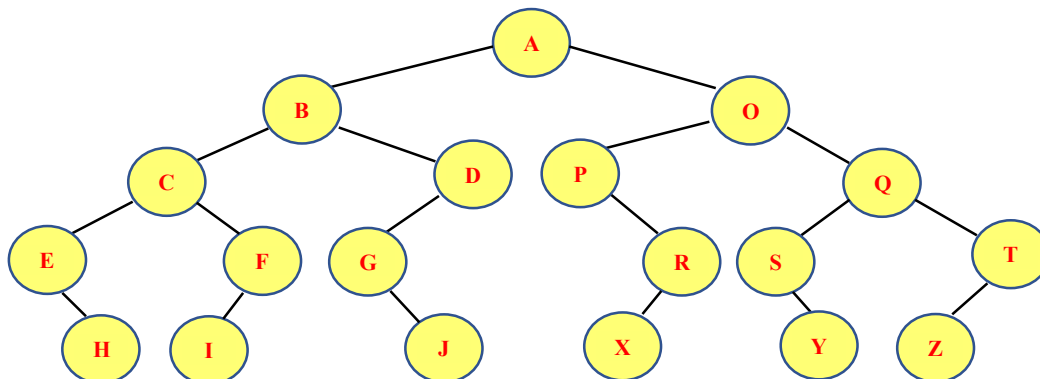
Bài 4: Vẽ cây nhị phân biểu diễn các biểu thức ở Bài 5. Kiểm tra lại kết quả Bài 5 bằng phương pháp duyệt cây LRN

Bài 5: Duyệt các cây nhị phân sau bằng các phương pháp NLR, NRL, LNR, RNL, LRN, RLN

1.



2.



Cho các dãy số

a) 8 2 3 9 6 5 1

b) 1 8 6 7 9 2 4 3 5

c) 7 3 4 5 6 2 1 8 9

d) 6 9 8 7 1 5 3 4 2

e) 1 2 3 4 5 6 7 8 9

f) 9 8 7 6 5 4 3 2 1

g) 1 3 5 7 9 2 4 6 8

h) 2 4 6 8 1 3 5 7 9

i) 1 2 9 8 3 4 7 6 5

j) 9 1 8 2 7 3 6 4 5

Bài 6: Xếp các dãy số trên vào cây nhị phân tìm kiếm. Hãy vẽ lại cây nếu xóa nút số 5

Bài 7: Xếp các dãy số trên vào cây nhị phân tìm kiếm cân bằng (cây AVL). Hãy vẽ lại cây nếu xóa nút số 5

Bài 8: Sắp xếp các dãy số trên theo thứ tự từ nhỏ đến lớn và từ lớn đến nhỏ theo các phương pháp Selection, Insertion, Bubble.

Bài 9: Sắp xếp các dãy số trên theo thứ tự từ nhỏ đến lớn và từ lớn đến nhỏ theo các phương pháp Quicksort, Heapsort, Mergesort (Mergesort và Natural Mergesort).

Bài 10: Xếp các dãy số trên vào bảng băm tìm kiếm kết nối trực tiếp và kết nối hợp nhất

BÀI TẬP THỰC HÀNH CẤU TRÚC DỮ LIỆU

TH 1 (Đề quy): Xây dựng hàm đệ quy bằng ngôn ngữ C các biểu thức sau:

1. $S = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$ với $n > 0$
2. $S = 1 * 3 * 5 * \dots * (2n + 1)$ với $n > 0$
3. $S = 1 - 2 + 3 - 4 + (-1)^{n+1}n$ với $n > 0$
4. $S = 1! + 2! + \dots + n!$ với $n > 0$
5. Đếm số lượng chữ số của một số nguyên n (gợi ý: $dem = 1 + n/10$)
6. Tìm ước số chung lớn nhất, bội số chung nhỏ nhất của 2 số nguyên a và b (USCLN: $a - b$)
7. Một người gửi số tiền X vào ngân hàng với lãi suất hằng năm là a với cơ chế lãi gộp vào gốc. Cứ sau mỗi 10 năm thì ngân hàng sẽ có một món quà nhỏ có giá trị b gộp vào tài khoản tiền đã gửi. Hỏi sau N năm thì người đó có bao nhiêu tiền trong tài khoản?
8. $x_n = (x_{n-1} + x_{n-2})^2 + (x_{n-1} - x_{n-2})^2$ với $x_0 = 0, x_1 = 1$
9. $x_n = 2x_{n-1} + 3y_{n-1}$, $y_n = 3x_{n-1} - 2y_{n-1}$ với $x_0 = 1, y_0 = 1$
10. $x_n = n!x_0 + (n-1)!x_1 + (n-2)!x_2 + \dots + 1!x_{n-1}$ với $x_0 = 1$
11. Cài đặt bài toán tìm đường đi của con chuột trong mê cung.

TH 2 (Danh sách liên kết – Stack): Viết chương trình ngôn ngữ C thực hiện những việc sau

1. Cài đặt hàm tạo danh sách liên kết, thêm phần tử (vị trí đầu tiên, cuối cùng, bất kỳ), xóa phần tử (vị trí đầu tiên, cuối cùng, bất kỳ).
2. Tạo danh sách liên kết đơn lưu trữ các số thực. Tìm số dương nhỏ nhất trong danh sách liên kết đơn đã tạo.
3. Tạo danh sách liên kết đơn lưu trữ các số thực. Tìm số âm lớn nhất trong danh sách liên kết đơn đã tạo.
4. Tạo danh sách liên kết đơn lưu trữ các số thực. Tính tổng các số dương trong danh sách liên kết đơn đã tạo.
5. Tạo danh sách liên kết đơn lưu trữ các số thực. Tính tổng các số âm trong danh sách liên kết đơn đã tạo.
6. Tạo danh sách liên kết đơn lưu trữ các số nguyên. Tìm số lẻ đầu tiên có trong danh sách liên kết đơn đã tạo.
7. Tạo danh sách liên kết đơn lưu trữ các số nguyên. Tìm số chẵn cuối cùng có trong danh sách liên kết đơn đã tạo.
8. Tạo danh sách liên kết đơn lưu trữ các ký tự. Đếm số ký tự 'A' có trong danh sách liên kết đơn đã tạo.
9. Tạo danh sách liên kết đơn lưu trữ các ký tự. Cho biết thứ tự của phần tử cuối cùng mang ký tự 'A'

10. Tạo danh sách liên kết đơn lưu trữ chuỗi ký tự có độ dài tối đa là 20. Cho biết thứ tự của phần tử có chuỗi ký tự ngắn nhất (phần tử đầu tiên có thứ tự là 1)
11. Tạo danh sách liên kết đơn lưu trữ chuỗi ký tự có độ dài tối đa là 20. In ra giá trị (chuỗi ký tự) của phần tử có chuỗi ký tự dài nhất (phần tử đầu tiên có thứ tự là 1).
12. Đảo ngược chuỗi ký tự sử dụng stack.
13. Chuyển đổi 1 số từ cơ số 10 (số tự nhiên) sang cơ số bất kỳ sử dụng stack.
14. Tính giá trị của 1 biểu thức hậu tố bất kỳ.
15. Chuyển đổi 1 biểu thức trung tố sang biểu thức hậu tố và ngược lại.
16. Dùng stack, viết chương trình phân tích 1 số thành thừa số nguyên tố theo thứ tự lớn trước nhỏ sau. Ví dụ $n = 3960$, hiển thị: $3960 = 11 * 5 * 3 * 3 * 2 * 2 * 2$
17. Mô phỏng CSDL, tổ chức 2 bảng dữ liệu SINHVIEN(masv, hosv, tensv, gioitinh, makhoa) và KHOA(makhoa, tenkhoa).

TH 3 (Cây nhị phân tìm kiếm – Cây AVL): Viết chương trình ngôn ngữ C thực hiện những việc sau

1. Tạo các cây nhị phân tìm kiếm lưu trữ các số nguyên, số thực, ký tự (so sánh theo thứ tự của chữ cái trong bảng ký tự).
2. Duyệt cây và in ra giá trị các nút của một cây ở câu 1
3. Đếm số nút chỉ có 1 nhánh của một cây nhị phân ở câu 1.
4. Tính chiều cao của một cây nhị phân ở câu 1
5. Tìm 1 nút có giá trị bất kỳ của một cây nhị phân ở câu 1. Tính mức và số nút lá của nút vừa tìm thấy.
6. Xóa 1 nút có giá trị bất kỳ của một cây nhị phân ở câu 1, sau đó tổ chức lại cây.
7. Sử dụng cây nhị phân tìm kiếm lưu trữ các số thực. Tính tổng giá trị của các nút nhỏ hơn một giá trị xác định trước.
8. Sử dụng cây nhị phân tìm kiếm lưu trữ các số nguyên. Đếm số lượng các nút có giá trị chẵn.
9. Sử dụng câu 2. xác định một cây ở câu 1 có cân bằng hay không?
10. Tạo cây nhị phân tìm kiếm cân bằng lưu trữ các số nguyên, số thực, ký tự (so sánh theo thứ tự của chữ cái trong bảng ký tự).

TH 4 (Sắp xếp): Viết chương trình ngôn ngữ C thực hiện những việc sau

(Dữ liệu đầu vào được đọc từ file)

1. Cài đặt các thuật toán sắp xếp Selection, Insertion, Bubble theo thứ tự từ nhỏ đến lớn và từ lớn đến nhỏ.
2. Cài đặt các thuật toán sắp xếp Quicksort, Heapsort, Mergesort (Mergesort và Natural Mergesort) theo thứ tự từ nhỏ đến lớn và từ lớn đến nhỏ.

3. Cho dãy ký tự được tổ chức thành nhiều từ, hãy sắp xếp các từ trong dãy theo thứ tự xác định.
4. Cho file dữ liệu danh sách sinh viên bao gồm Mã số sinh viên, Họ tên sinh viên, Điểm TB ví dụ như sau:

65123456 Nguyễn Văn Xuân 7.2

65987654 Trần Lan 6.6

....

Hãy sắp xếp danh sách sinh viên theo các tiêu chí được lựa chọn (mã số, họ, tên, điểm TB) và ghi vào file output.

5. Cho dãy số gồm n số thực, sử dụng các thuật toán đã cài đặt sắp xếp các phần tử của dãy đó với các yêu cầu sau:

- Từ trái qua phải:
 - Các số dương xuất hiện theo giá trị tăng dần.
 - Các số âm xuất hiện theo giá trị giảm dần.
- Không thay đổi vị trí của phần tử mang giá trị 0.
- Không thay đổi tính chất ở mỗi vị trí (nghĩa là nếu trước khi sắp xếp vị trí i có giá trị âm thì sau khi sắp xếp vị trí i cũng phải mang giá trị âm, nếu vị trí i mang giá trị dương cũng tương tự).

Các phần tử được ghi vào file (input và output) cách nhau 1 khoảng trắng.

Ví dụ:

Input	Output
6 3 -1 2 0 -4 6	2 -1 3 0 -4 6

TH 4 (Tìm kiếm): Viết chương trình ngôn ngữ C thực hiện những việc sau

1. Cài đặt hàm tổ chức và tìm kiếm tuần tự, nhị phân, bảng băm kết nối trực tiếp (kết nối ngoài), bảng băm kết nối hợp nhất (kết nối trong).
2. Sử dụng các thuật toán tìm kiếm để tìm kiếm số âm đầu tiên trong mảng.
3. Sử dụng các thuật toán tìm kiếm để tìm kiếm số dương đầu tiên trong mảng.
4. Hoán đổi phần tử nhỏ nhất đầu tiên với phần tử lớn nhất cuối cùng.
5. Có n vé đi xem ca nhạc có sẵn, mỗi vé có 1 giá bán nhất định. Sau đó có m người khách hàng đến mua vé, lần lượt từng người một. Mỗi người ra giá cao nhất mà họ có thể mua, sau đó họ sẽ nhận được một vé có giá lớn nhất sao cho không vượt quá giá tối đa. Hãy cho biết giá của mỗi vé mà mỗi người mua được.