

BÀI TẬP 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);
```

```
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);
```

```
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);
```

```
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);
```

```
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);
```

```
6. 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);
```

```
7. 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);
```

```
8. 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: 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 $x_0 = 1$

Bài 3: Viết chương trình ngôn ngữ C thực hiện những việc sau

1. Tạo danh sách liên kết đơn lưu trữ các số thực. Tìm số dương nhỏ nhất trong danh sách liên kết đơn đã tạo.
2. Tạo danh sách liên kết đơn lưu trữ các số thực. Tìm số âm lớn nhất trong danh sách liên kết đơn đã tạo.
3. Tạo danh sách liên kết đơn lưu trữ các số thực. Tính tổng các số dương trong danh sách liên kết đơn đã tạo.
4. Tạo danh sách liên kết đơn lưu trữ các số thực. Tính tổng các số âm trong danh sách liên kết đơn đã tạo.
5. 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.
6. 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.
7. 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.
8. Tạo danh sách liên kết đơn lưu trữ các ký tự. Cho biết thứ tự của phần tử cuối cùng mang ký tự 'A'
9. 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)
10. 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)

Bài 4: Tính giá trị các biểu thức hậu tố sau:

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

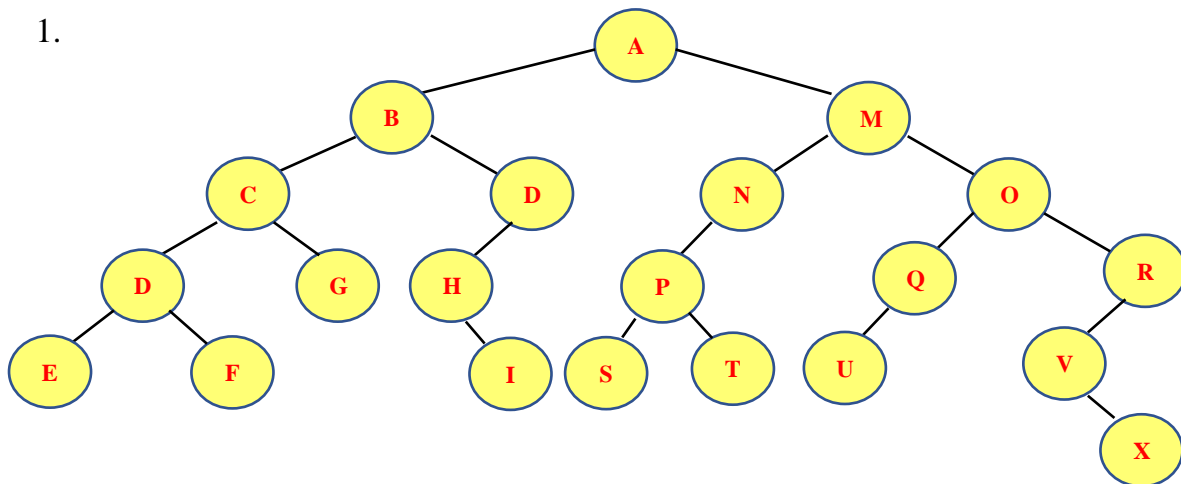
Bài 5: Chuyển đổi các biểu thức sau thành biểu thức hậu tố bằng phương pháp sử dụng Stack

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

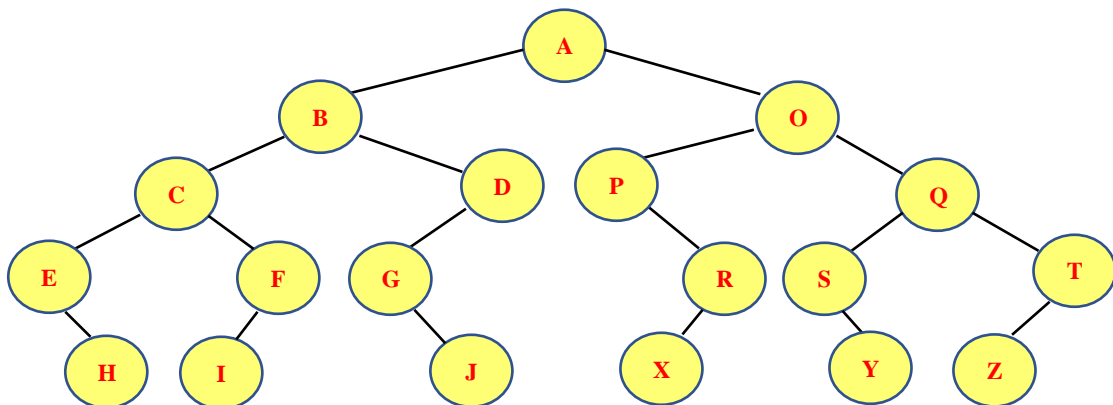
Bài 6: Vẽ cây nhị phân biểu diễn các biểu thức ở **Bài 5**. Kiểm tra lại kết quả Bài 5 bằng phương pháp duyệt cây LRN

Bài 7: 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 8: Xếp các dãy số trên vào cây nhị phân tìm kiếm. Hãy vẽ lại cây nếu xóa nút số 5

Bài 9: Xếp các dãy số trên vào cây nhị phân tìm kiếm cân bằng (cây AVL). Hãy vẽ lại cây nếu xóa nút số 5