

Đề thi CẤU TRÚC DỮ LIỆU VÀ GIẢI THUẬT
 Học kỳ I, 2015-2016
 Thời gian: 95 phút

Lưu ý: Đề thi có 02 tờ. Thí sinh không được sử dụng tài liệu.

Câu 1 (1 điểm)

Tính độ phức tạp tính toán cho hàm sum() dưới đây:

```
int sum(int m, int n, int K ) {
    int s = 0;
    for (int i = 0; i < m; i ++) {
        s += i;
        if (i == K) {
            for (int j = 0; j < n; j ++) {
                s += j;
            }
        }
    }
    return s;
}
```

Câu 2 (1 điểm)

Cho một dãy A gồm n nguyên dương được lưu giữ dưới dạng một danh sách liên kết đơn (single linked-list). Mỗi nút trong danh sách có cấu trúc như sau:

<pre>// C/C++ Node { int data; Node* next; }</pre>		<pre>//Java class Node { int data; Node next; }</pre>
--	--	---

Hãy viết hàm `Node* find(Node* head, int key)` với C/C++, hay `Node find(Node head, int key)` với Java để tìm xem phần tử có khoá `key` có trong dãy A hay không? Trả về phần tử đó nếu có, còn nếu không trả về `null` (rỗng).

Câu 3 (1.5 điểm)

Giả sử có hàng đợi (queue) Q gồm các giá trị $[4, 3, 2, 1]$ (trong đó 1 ở đầu hàng đợi (bên phải) có thể lấy ra, và 4 ở đuôi hàng đợi (bên trái) nơi có thể thêm tiếp vào); và có một ngăn xếp (stack) S rỗng.

Chỉ sử dụng S , Q và thêm một biến duy nhất nếu cần (không được sử dụng thêm bất kỳ cấu trúc dữ liệu nào khác), hãy liệt kê các thao tác để có được ngăn xếp S như sau:

a. $[1, 2]$ b. $[2, 4]$ c. $[2, 1]$

(Trong đó đầu ngăn xếp ở bên phải, tức là 2 trong câu (a), và 1 trong câu (c)).

Câu 4 (1 điểm)

Trình bày ngắn gọn thuật toán sắp xếp nhanh (Quick sort). Minh hoạ bằng việc sắp xếp theo thứ tự tăng dần dãy số sau: 7, 6, 10, 8, 9, 4, 3, 5. (Nêu rõ cách chọn phần tử chốt (pivot) khi trình bày.)

Câu 5 (0.5 điểm)

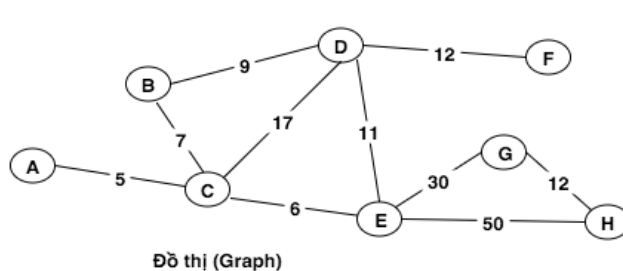
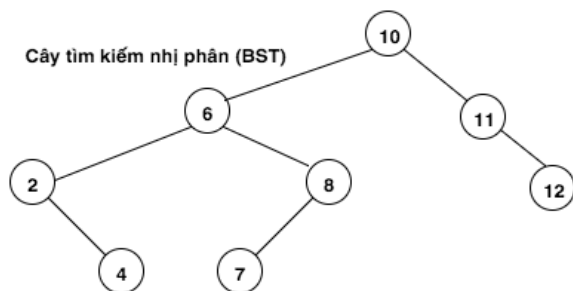
Độ cao nhỏ nhất của cây nhị phân có 15 đỉnh là bao nhiêu? Hãy vẽ cây nhị phân có độ cao nhỏ nhất này.

Câu 6 (1.5 điểm)

Cho cây tìm kiếm nhị phân như trong hình vẽ (ở dưới, bên trái). Trình bày các bước cho việc:

a. Thêm đỉnh có khoá 9 vào cây trên.

b. Xoá đỉnh có khoá 6 ra khỏi cây trên.

**Câu 7** (1.5 điểm)

Cho đồ thị vô hướng có trọng số như trong hình vẽ (ở trên, bên phải).

a. Xuất phát từ đỉnh A, hãy chỉ ra thứ tự các đỉnh sẽ được thăm khi tiến hành duyệt trên đồ thị theo chiều sâu (DFS - depth first search). (Thứ tự duyệt các đỉnh liền kề theo trật tự bảng chữ cái.)

b. Sử dụng một trong hai thuật toán Prim hoặc Kruskal, hãy tìm cây bao trùm nhỏ nhất (minimum spanning tree) cho đồ thị trên. Nêu ra kết quả trung gian khi trình bày.

Câu 8 (1 điểm)

Cho dãy A gồm n số nguyên trong đoạn từ 1 tới 100. Hãy trình bày thủ tục đếm số lần xuất hiện của mỗi số nguyên có trong dãy A. Đánh giá độ phức tạp tính toán. (Thuật toán càng nhanh thì càng được điểm cao).

Câu 9 (1 điểm)

Cho một số nguyên dương n bất kỳ. Chỉ dùng các phép toán cơ bản: cộng, trừ, nhân, chia, so sánh và các phép toán logic, hãy viết thuật toán xác định xem số nguyên dương n có phải là số chính phương hay không (tức $n=q*q$ trong đó q nguyên)? Đánh giá độ phức tạp tính toán. (Thuật toán càng nhanh thì càng được điểm cao).

Câu 10 (Câu điểm thưởng, và có thể bỏ qua không làm; 1 điểm)

Một danh sách liên kết có chu trình (vòng lặp) nếu có ít nhất một node được thăm lại khi duyệt tuần tự. Ví dụ :

Danh sách không có chu trình:
1 --> 2 --> 3 --> null

Danh sách có chu trình:

1 --> 2 --> 3
 ^ |
 | |

Cho con trỏ chỉ tới đầu một danh sách liên kết. Hãy trình bày ý tưởng và các bước chính để xác định xem danh sách này có chu trình hay không, với thời gian $O(n)$ và bộ nhớ $O(1)$ (tức ngoài bộ nhớ đang dùng cho danh sách liên kết, chỉ được dùng thêm không quá C biến, trong đó C là một số cố định cho trước) ?