

CẤU TRÚC DỮ LIỆU
Học kỳ 3 nhóm 2 - 1718

Thời gian: 90 phút

Họ và tên: _____

1 Phân thi trắc nghiệm (5 điểm)

1. (0.35 đ)Hãy tính độ phức tạp tính toán của đoạn code sau:

```
void silly(int n) {
    if (n <= 0) return;
    cout<<"n = "<< n;
    silly(n-1);
}
```

- (a) $\mathcal{O}(n^3)$ (b) $\mathcal{O}(n \log n)$ (c) $^* \mathcal{O}(n)$ (d) $\mathcal{O}(\log n)$

2. (0.35 đ)Hãy tính độ phức tạp tính toán của đoạn code sau:

```
void silly(int n) {
    if (n <= 0) return;
    cout<<"n = "<< n;
    silly(n/2);
}
```

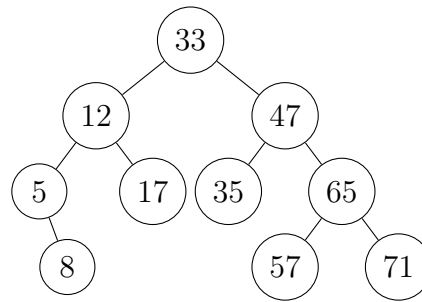
- (a) $\mathcal{O}(n \log n)$ (b) $\mathcal{O}(n^2)$ (c) $^* \mathcal{O}(\log n)$ (d) $\mathcal{O}(n^2 \log n)$

3. (0.35 đ)Hãy tính độ phức tạp tính toán của đoạn code sau:

```
int silly(int n, int m) {
    if (n < 1) return m;
    else if (n < 10)
        return silly(n/2, m);
    else
        return silly(n - 2, m);
}
```

- (a) $\mathcal{O}(n \log n)$ (b) $\mathcal{O}(n^2)$ (c) $\mathcal{O}(\log n)$ (d) $^* \mathcal{O}(n)$

4. (0.35 đ) Cây sau có phải là cây cân bằng AVL hay không?



- (a) * Có (b) Không (c) (d)

5. (0.35 đ) Thứ tự duyệt tiền tố của đồ thị trên là

- (a) 5, 8, 12, 17, 33, 35, 47, 57, 65, 71 (b) 8, 5, 17, 12, 35, 57, 71, 65, 47, 33
(c) * 33, 12, 5, 8, 17, 47, 35, 65, 57, 71 (d) 33, 12, 47, 5, 17, 35, 65, 8, 57, 71

6. (0.35 đ) Thứ tự duyệt trung tố của đồ thị trên là

- (a) * 5, 8, 12, 17, 33, 35, 47, 57, 65, 71 (b) 8, 5, 17, 12, 35, 57, 71, 65, 47, 33
(c) 33, 12, 5, 8, 17, 47, 35, 65, 57, 71 (d) 33, 12, 47, 5, 17, 35, 65, 8, 57, 71

7. (0.35 đ) Thứ tự duyệt hậu tố của đồ thị trên là

- (a) 5, 8, 12, 17, 33, 35, 47, 57, 65, 71 (b) * 8, 5, 17, 12, 35, 57, 71, 65, 47, 33
(c) 33, 12, 5, 8, 17, 47, 35, 65, 57, 71 (d) 33, 12, 47, 5, 17, 35, 65, 8, 57, 71

8. (0.35 đ) Thứ tự duyệt BFS của đồ thị trên là

- (a) 5, 8, 12, 17, 33, 35, 47, 57, 65, 71 (b) 8, 5, 17, 12, 35, 57, 71, 65, 47, 33
(c) 33, 12, 5, 8, 17, 47, 35, 65, 57, 71 (d) * 33, 12, 47, 5, 17, 35, 65, 8, 57, 71

9. (0.35 đ) Hãy cho biết chức năng của hàm sau biết rằng head là một con trỏ đầu của một LinkedList:

```

void fun1(node* head)
{
    if(head == NULL)
        return;

    fun1(head->next);
    cout<<head->data<<" ";
}
    
```

- (a) In tất cả phần tử của LinkedList
(b) * In tất cả phần tử của LinkedList theo thứ tự đảo ngược
(c) In tất cả phần tử thay thế của LinkedList
(d) In tất cả phần tử thay thế của LinkedList theo thứ tự đảo ngược

10. (0.4 đ) Which of the following points are true about Linked List data structure when it is compared with array:
- (a) Arrays have better cache locality that can make them better in terms of performance
 - (b) It is easy to insert and delete elements in Linked List
 - (c) Random access is not allowed in a typical implementation of Linked Lists
 - (d) * All of the above

11. (0.4 đ) Thuật toán nào dưới đây có thể được sử dụng để sắp xếp một Linkedlist với $\mathcal{O}(\text{BigO})$ tốt nhất:

(a) Insertion Sort (b) Quick Sort (c) Heap Sort (d) * Merge Sort

12. (0.35 đ) What is the output of following function for start pointing to first node of following linked list? $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6$

```
void fun(struct node* start)
{
    if(start == NULL)
        return;
    cout<<start->data<<" ";

    if(start->next != NULL )
        fun(start->next->next);
    cout<<start->data<<" ";
}
```

(a) 1 4 6 6 4 1 (b) 1 3 5 1 3 5 (c) * 1 3 5 5 3 1 (d) 1 2 3 5

13. (0.35 đ) Xét hàm sau được dùng để duyệt LinkedList. Điều nào dưới đây là sai khi nói về hàm này?

```
void traverse(Node *head)
{
    while (head->next != NULL)
    {
        cout<<head->data<<" ";
        head = head->next;
    }
}
```

- (a) Hàm có thể treo khi LinkedList trống
- (b) Hàm không thể in ra phần tử cuối cùng của LinkedList
- (c) * Hàm cài đặt không đúng vì nó làm thay đổi head
- (d)

14. (0.35 đ) Cho P là một danh sách liên kết đơn(singly linked list). Và Q là con trỏ tới nút x trong danh sách P. Hỏi độ phức tạp thời gian xấu nhất của thuật toán để xóa nút x khỏi danh sách là gì?

(a) $\mathcal{O}(n)$ (b) $\mathcal{O}(\log 2n)$ (c) $\mathcal{O}(\log n)$ (d) * $\mathcal{O}(1)$

