

# DSA Assignment 3

Nguyễn Thành Phát - 22028108

# Mục lục

1	Bài 1	2
2	Bài 2	2
3	Bài 3	2
4	Bài 4	3
5	Bài 5	3
6	Bài 6	4

## 1 Bài 1

Đề bài: Sắp xếp các hàm số dưới theo thứ tự độ phức tạp  $\mathcal{O}$

$$2^{10} < 2^{\log n} = n < 4n < 3n + 100 \log n < n \log n < 4n \log n + 2n < n^2 + 10n < n^3 < 2^n$$

## 2 Bài 2

Đề bài: Cho một số nguyên  $n$ , viết 2 thuật toán để tính  $2^n$  bằng mã giả.

**Mã giả cho thuật toán 1:**

```
result ← 1
for  $i = 1 \rightarrow n$  do
    result ← result * 2
end for
return result
```

Độ phức tạp của thuật toán trên là  $O(n)$

**Mã giả cho thuật toán 2:**

```
result ← 1
base ← 2
for  $n \neq 0$  do
    if  $n$  is odd then
        result ← result * base
    else
        base ← base * base
    end if
     $n \leftarrow n/2$ 
end for
return result
```

Độ phức tạp của thuật toán trên là  $\mathcal{O}(\log n)$

## 3 Bài 3

Đề bài: Viết mã giả và phân tích độ phức tạp của các phương thức của hàng đợi (queue) cài đặt bằng mảng tĩnh.

Bản cài đặt của bài này viết bằng C++ ở đây: [Queue\\_AR.cpp](#)

Độ phức tạp của các phương thức có trong cấu trúc dữ liệu trên:

- Hàm *enqueue*:  $\mathcal{O}(1)$
- Hàm *dequeue*:  $\mathcal{O}(1)$
- Hàm *front*:  $\mathcal{O}(1)$
- Hàm *size*:  $\mathcal{O}(1)$
- Hàm *empty*:  $\mathcal{O}(1)$

## 4 Bài 4

Đề bài: Viết mã giả và phân tích độ phức tạp của các phương thức của hàng đợi (queue) cài đặt bằng danh sách liên kết.

Bản cài đặt của bài này viết bằng C++ ở đây: [Queue\\_LL.cpp](#)

Độ phức tạp của các phương thức có trong cấu trúc dữ liệu trên:

- Hàm *enqueue*:  $\mathcal{O}(n)$
- Hàm *dequeue*:  $\mathcal{O}(1)$
- Hàm *front*:  $\mathcal{O}(1)$
- Hàm *size*:  $\mathcal{O}(1)$
- Hàm *empty*:  $\mathcal{O}(1)$

## 5 Bài 5

Đề bài: Viết mã giả và phân tích độ phức tạp của các phương thức của ngăn xếp (stack) cài đặt bằng mảng tĩnh.

Bản cài đặt của bài này viết bằng C++ ở đây: [Stack\\_AR.cpp](#)

Độ phức tạp của các phương thức có trong cấu trúc dữ liệu trên:

- Hàm *push*:  $\mathcal{O}(1)$
- Hàm *pop*:  $\mathcal{O}(1)$
- Hàm *top*:  $\mathcal{O}(1)$
- Hàm *size*:  $\mathcal{O}(1)$
- Hàm *empty*:  $\mathcal{O}(1)$

## 6 Bài 6

Đề bài: Viết mã giả và phân tích độ phức tạp của các phương thức của ngăn xếp (stack) cài đặt bằng danh sách liên kết.

Bản cài đặt của bài này viết bằng C++ ở đây: [Stack\\_LL.cpp](#)

Độ phức tạp của các phương thức có trong cấu trúc dữ liệu trên:

- Hàm *push*:  $\mathcal{O}(1)$
- Hàm *pop*:  $\mathcal{O}(1)$
- Hàm *top*:  $\mathcal{O}(1)$
- Hàm *size*:  $\mathcal{O}(1)$
- Hàm *empty*:  $\mathcal{O}(1)$