

**Bài 1:**  $W(n)$ ,  $A(n)$  lần lượt là thời gian chạy trường hợp xấu nhất và trung bình của thuật toán có kích thước là  $n$ . Điều nào dưới đây luôn đúng?

- (A)  $A(n) = \Omega(W(n))$
- (B)  $A(n) = \Theta(W(n))$
- (C)  $A(n) = O(W(n))$
- (D)  $A(n) = o(W(n))$

Chọn đáp án C

**Bài 2:** Độ phức tạp của hàm `fun()`?

```
int fun(int n)
{
    int count = 0;
    for (int i = n; i > 0; i /= 2)
        for (int j = 0; j < i; j++)
            count += 1;
    return count;
}
```

Độ phức tạp:  $O(n + n/2 + n/4 + \dots + 1) = O(n)$

**Bài 3:** Độ phức tạp của hàm đệ quy sau là bao nhiêu?

```
int DoSomething (int n)
{
    if (n <= 2)
        return 1;
    else
        return (DoSomething (floor(sqrt(n))) + n);
}
```

Ta có,  $T(n) = T(\sqrt{n}) + C$

$T(n) = T(n^{1/2^k}) + kC$

$n^{1/2^k} = 2$

Do đó,  $T(n) = O(\log \log(n))$

**Bài 4:** Độ phức tạp xấu nhất khi thực hiện đoạn chương trình?

```
bool isSubsetSum(int set[], int n, int sum)
{
    // Base Cases
    if (sum == 0)
        return true;
    if (n == 0 && sum != 0)
        return false;

    // If last element is greater than sum, then ignore it
    if (set[n-1] > sum)
        return isSubsetSum(set, n-1, sum);

    /* else, check if sum can be obtained by any of the following
       (a) including the last element
       (b) excluding the last element */
    return isSubsetSum(set, n-1, sum) ||
           isSubsetSum(set, n-1, sum-set[n-1]);
}
```

---