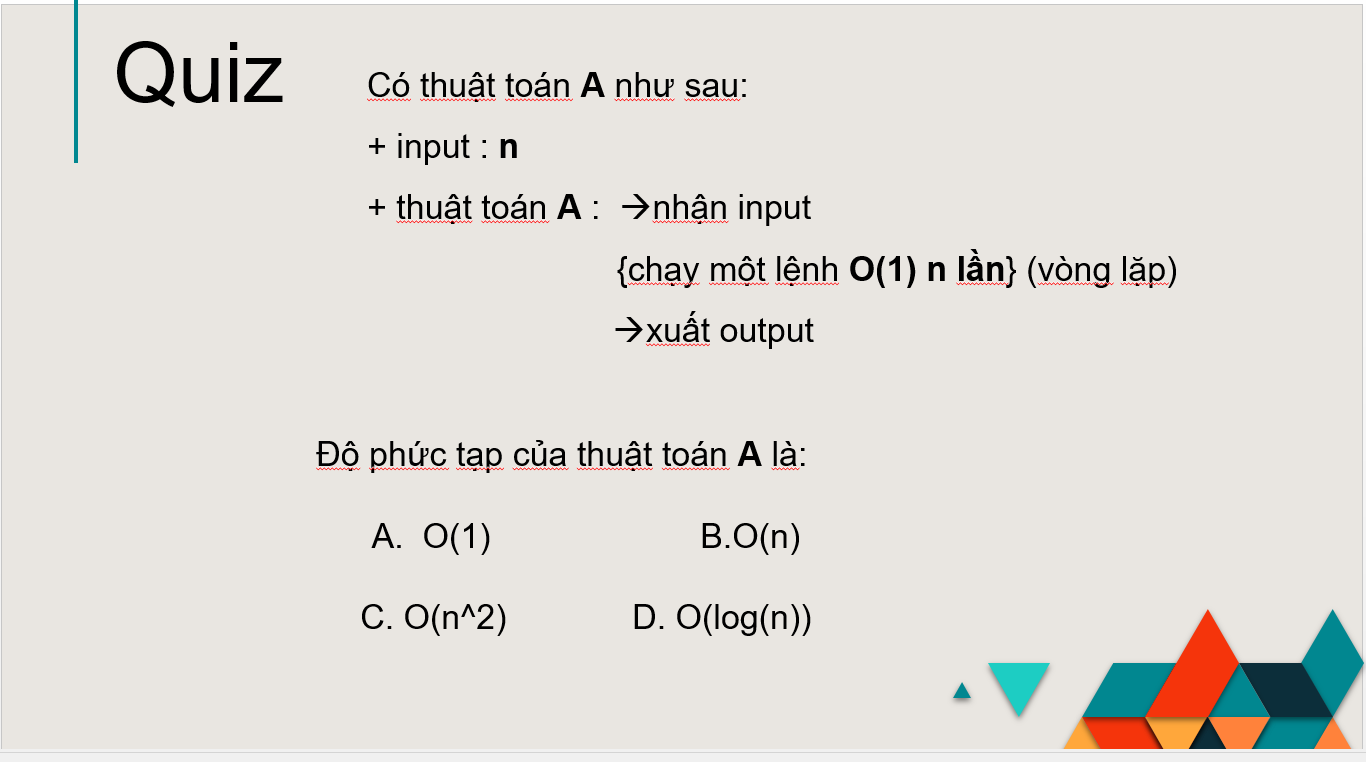
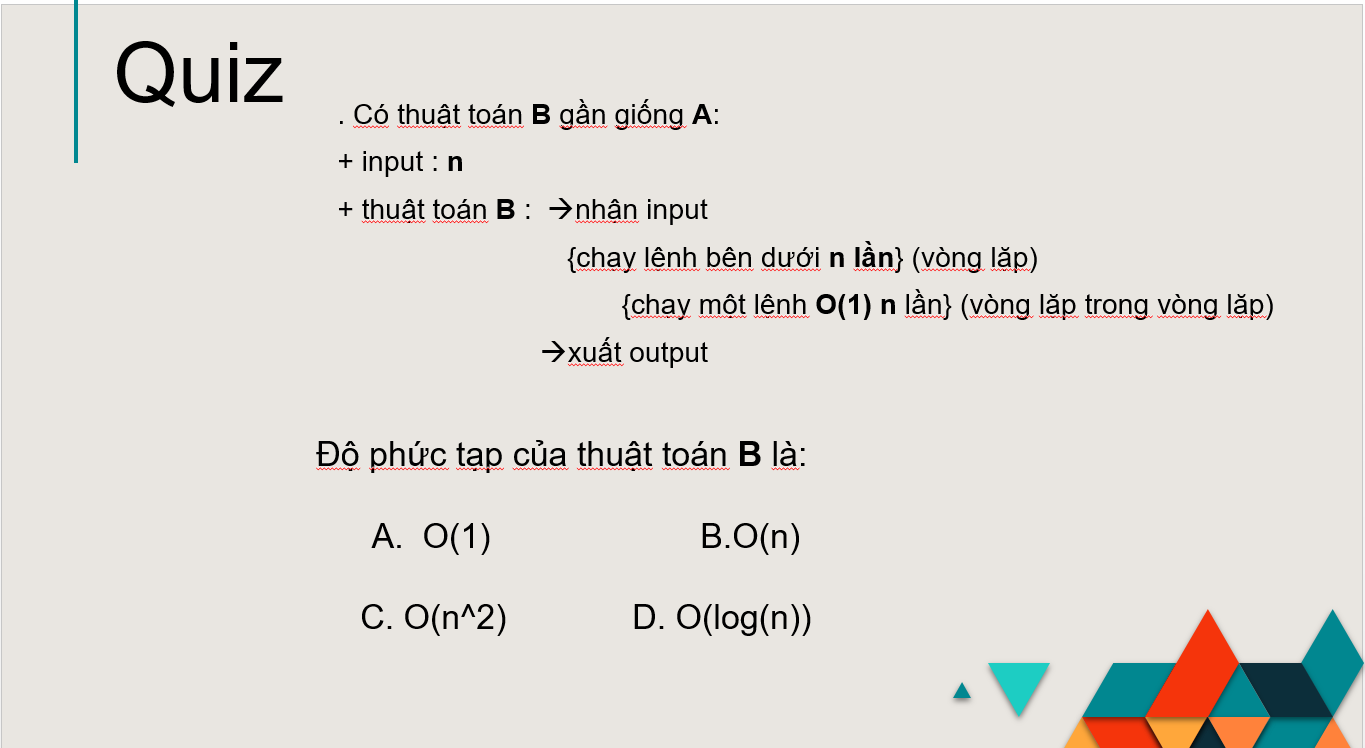
**QUÁ TRÌNH TƯƠNG TÁC CỦA NHÓM VÀ LỚP TRONG BUỔI THUYẾT TRÌNH**

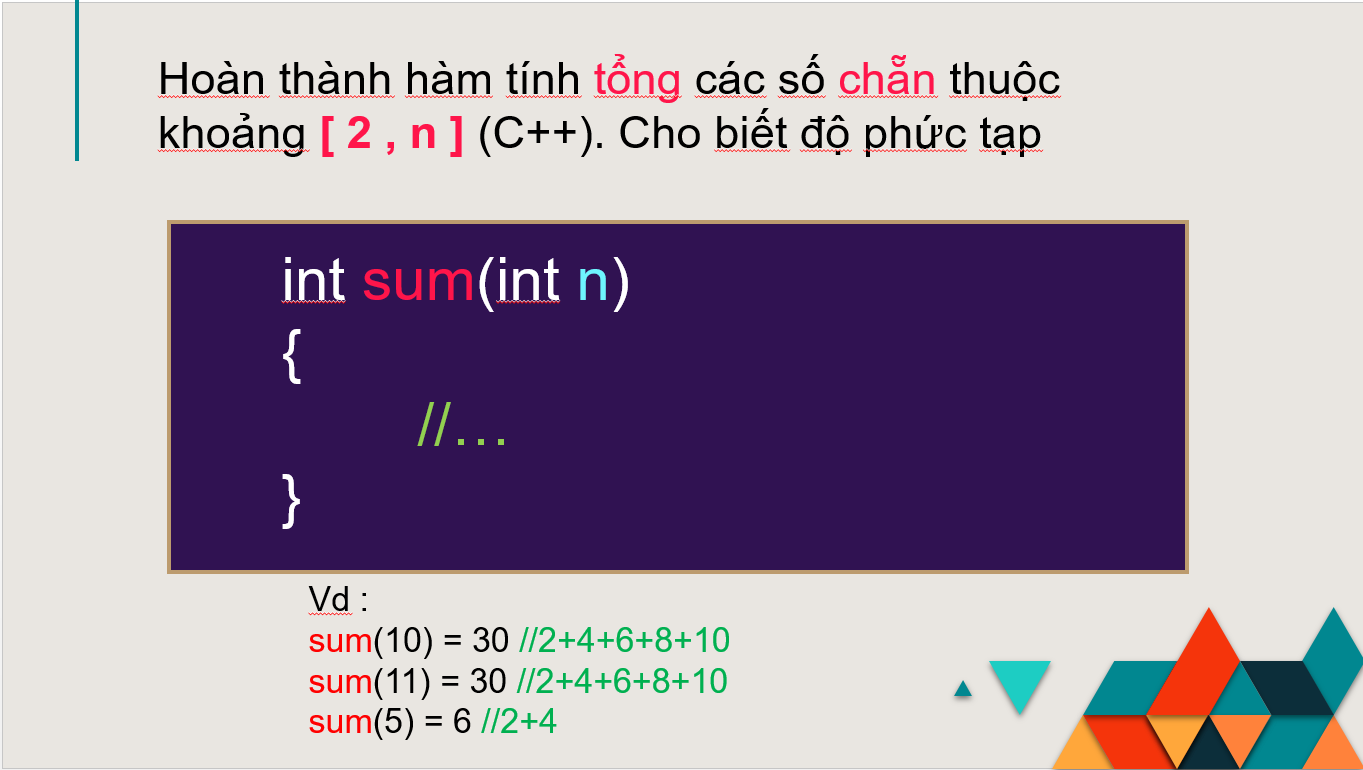
* **Những câu hỏi nhóm đặt cho lớp:**
* **Thuật toán là gì?**
* Nhóm 6: Thuật toán là ý tưởng để giải quyết một bài toán nào đó.
* Nhóm 5: Thuật toán là tập hợp nhiều thao tác được thực hiện theo trình tự nhằm hoàn thành mục đích đã xác định trước.
* **Thuật toán cần có những tính chất gì?**
* Nhóm 11, 15: Tính chính xác, rõ ràng, đơn giản, khách quan, hiệu quả.
* **Những cách biểu diễn thuật toán?**
* Nhóm 10: Dùng lưu đồ, mã giả.
* **Có những loại thuật toán nào?**
* Nhóm 9: Quy hoạch động, chia để trị, đệ quy.
* **Thuật toán có cần thiết không?**
* Nhóm 11: Khi biết nhiều thuật toán chúng ta dễ dàng có ý tưởng để giải quyết các vấn đề.
* Nhóm 4: Đỡ tốn thời gian, dễ sử dụng.
* **Một thuật toán được đánh giá qua các tiêu chí nào?**
* Nhóm 10: Thời gian chạy và dữ liệu tiêu hao của thuật toán.
* **Tỉ suất tăng là gì?**
* Nhóm 6: Khi dữ liệu đầu vào của một chương trình đủ lớn, với tỉ suất tăng chúng ta sẽ tìm ra được thời gian chạy của chương trình.
* **Big O là gì?**
* Nhóm 17: Big O là khoảng thời gian tiêu hao để chạy một thuật toán, Big O được sử dụng để so sánh độ hiệu quả của các thuật toán.
* **Quiz:**
* Quiz 1: Nhóm 7.

****

* Quiz 2: Nhóm 3.

****

* Quiz 4:

****

Nhóm 4:

int r=0;

for(int i=2; i<=n; i+=2){

r+=i;

return r;

}

* O(n)

Nhóm 11:

* 0(1)
* **Một số câu hỏi nhóm nhận được:**

1. Thuật toán có nhiều vòng lặp thì độ phức tạp sẽ được tính như thế nào? – Nhóm 1

* Độ phức tạp của thuật toán sẽ được tính dựa trên quy tắc cộng và quy tắc nhân.

+ QT cộng: Nếu các vòng lặp riêng biệt với nhau thì độ phức tạp sẽ được tính bằng tổng thời gian chạy của các vòng lặp.

+ QT nhân: Nếu các vòng lặp lồng vào nhau thì độ phức tạp sẽ được tính bằng tích thời gian chạy của các vòng lặp.

1. Thế nào là một thuật toán hiệu quả? – Nhóm 13

* Thuật toán có độ phức tạp lưu trữ và độ phức tạp tính toán càng nhỏ thì càng hiệu quả.

1. Tỉ suất tăng là gì? – Nhóm 15

* Tỉ suất tăng Bigθ là tốc độ tăng trưởng của hàm số f(n), một thuật toán có tỉ suất tăng càng nhỏ thì càng hiệu quả.

1. Tại sao độ phức tạp của thuật toán Quick Sort là O(nlog(n)) mà không phải là O(n^2)? – Nhóm 6

* Vì độ phức tạp của Quick Sort là O(n^2) chỉ khi gặp trường họp xấu nhất (worst case), mà gần như không gặp trường hợp worst case. Vì với thuật toán Quick Sort, worst case chỉ xảy ra khi mảng đưa vào là một mảng đã sắp xếp sẵn, người ta thường có cách check xem mảng đã sắp xếp sẵn chưa trước khi được vào thuật toán Quick Sort. Nên độ phức tạp của thuật toán Quick Sort thường là O(nlog(n)).