1、什么叫数据结构？它在程序设计过程中有什么用处？

数据结构是一种组织和存储数据的方式，它涉及到数据的表示、存储和操作。

在程序设计过程中，数据结构有很重要的作用，它能够帮助程序员有效地管理和操作数据。以下是数据结构在程序设计中的几个常见用途：

1. 存储数据：数据结构提供了不同的方式来存储数据，如数组、链表、树、图等。不同的数据结构适合不同的应用场景，程序员可以根据实际需求选择最适合的数据结构。
2. 提高算法效率：选择合适的数据结构能够提高算法的效率，减少程序运行时间和空间复杂度。例如，使用哈希表可以提高查找和插入的效率，使用堆可以实现高效的排序和优先级队列操作。
3. 简化程序设计：合理的数据结构设计能够使程序的代码更简洁、易于理解和维护。例如，使用链表可以方便地实现插入和删除操作，使用树可以实现高效的搜索和遍历操作。
4. 支持复杂应用：一些复杂的应用，如数据库、编译器、操作系统等，需要处理大量的数据和复杂的数据结构。程序员需要熟悉不同的数据结构，才能够完成这些复杂的应用开发。

2、什么叫算法？如何进行算法分析？

算法是指用来解决问题或完成任务的一组有限的、确定的指令集合。在计算机科学中，算法通常用来解决计算问题，如搜索、排序、最短路径等。

算法的好坏决定了解决问题的效率和准确性，因此，需要对算法进行分析。算法分析的目的是评估算法的时间复杂度和空间复杂度，以便选择最优算法或优化算法。

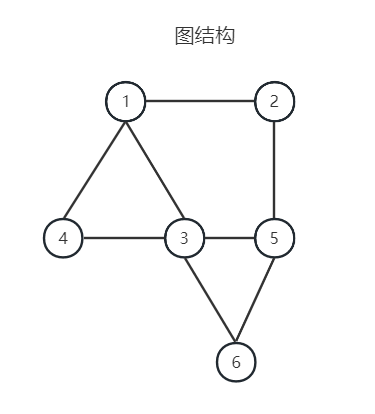
算法分析通常包括以下步骤：

1. 定义问题：明确问题的输入、输出和要求。
2. 设计算法：根据问题定义，设计一个解决问题的算法。
3. 衡量算法效率：通过计算算法的时间复杂度和空间复杂度，衡量算法的效率和资源消耗。
4. 评估算法优劣：比较不同算法的时间复杂度和空间复杂度，评估算法的优劣和适用性。

常用的算法分析方法有：

* 时间复杂度分析：计算算法的运行时间和输入规模之间的关系，通常用大 O 表示法表示。
* 空间复杂度分析：计算算法所需的额外空间和输入规模之间的关系。
* 最坏情况分析：分析算法在最坏情况下的时间复杂度，确保算法的性能不会太差。
* 平均情况分析：分析算法在不同输入情况下的时间复杂度的平均值。
* 通过算法分析，程序员能够选择最优算法，提高程序的效率和性能。

3、假设有数据结构（D，R），其中D={1,2,3,4,5,6}，R={（1，2），（1，4），（2，3），（2，4），（3，4），（3，5），（3，6），（4，5），（5，6））}。试画出其逻辑结构图并指出属于何种结构



4、(1) O(n³) (2) O(1)

5、(1) O(1) (2) O(n)