

1. **Дати визначення паралельної обробки. Недоліки та як їх позбутися.** Под режимом *параллельной обработки* данных нескольких задач понимается такой многопрограммный режим, в котором переход от одной задачи к другой происходит через достаточно короткие промежутки времени (кванты), сравнимые с тактом машины, чтобы создать у пользователя впечатление одновременности исполнения нескольких программ (режим кажущегося совмещения). Основной целью данного режима является улучшение обслуживания пользователей, выражающееся в том, что у последних создается впечатление отсутствия очереди, так как решение их задач не прерывается на длительные отрезки времени. Кроме того, если пользователю предоставляются некоторые средства прямого доступа (хотя бы для вывода информации), то это впечатление еще более усиливается выдачей результатов по мере их получения. Параллельная обработка основана на значительном отличии времени реакции пользователя и машины вследствие высокой скорости работы последней.
2. **Дати визначення процесу. Стратегія визначення пріоритетів в ОС.** Это любая выполняемая работа в системе; это динамический объект (внутренняя единица работы) системы, которому она выделяет ресурсы.
3. **Зв'язок модулів за керуванням. Які операції виконуються й якими програмами.** Связь по данным может быть через общие ресурсы либо через адрес списка параметров. Спомощью макрокоманд возможно указание:
 - возврат из (i+1)-го модуля только в i-й;
 - возврат из (i+n)-го модуля в i-й;
 - вызывающий модуль после передачи управления стираетсяДля связи по управлению данными определены некоторые ресурсы
4. **Особливості розв'язання задач в ОС ЛВС**
5. **Види модулів. Які програми беруть участь в перетворенні модулів .** Вихідний, об'єктний, завантажувальний, абсолютний. Компілятор між вихідним та об'єктним, редактор зв'язку між об'єктним та завантажувальним, завантажувач між завантажувальним та абсолютним
6. **Дати визначення табличного методу керування. Як формується таблиця векторів переривань.** принимается на основании инфы, хранящейся в таблицах, содержащих данные состояния всех частей системы. Решение обрабатывается не по анализу, а по состоянию таблиц, где оно фиксировано. Используется при старте машины
7. **Як визначити об'єм пам'яті, необхідної для завантаження програми динамічної паралельної структури** Корневой сегмент + наибольший суммирующий объем модулей, которые могут выполняться параллельно
8. **Чому алгоритм «Корбата» краще FBn** Алгоритм Корбата лучше алгоритма FBn, т.к. потенциально обладает большей производительностью т.к. распределяет задачи по очередям не по их приоритетам, а по признаку - абсолютная длина кода программы (чем меньше код тем выше приоритет). (В систему добавлен анализатор, который сразу размещает заявки в свою очередь, соответственно среднее время ожидания уменьшается.)
9. **Що таке критичні вершини .** Вершина, которая является последней в графе, она определяет время решения

- 10. Як операції виконує програма другого рівня планування** Второй уровень планирования определяет, какие процессы можно хранить в памяти, а какие — на диске. Этим занимается *планировщик памяти*
- 11. Як змінюється організація обчислювального процесу для SMP і PC**
- 12. Ідея оптимізації «базового» рішення** . «Базовое» решение оптимизируется по необходимому критерию - время выполнения, количество процессоров
- 13. Яка характеристика покращується в змішаних алгоритмах обслуговування** Среднее время ожидания заявки. Для ускорения работы системы
- 14. Дати визначення NP – повним завданням** Цей термін іде з динамічного планування і позначає складну задачу, яку можна вирішити, а саме $O(K^n)$
- 15. Різниця між плануванням і розподілом** Задача — распределение (загрузка) (TaskAllocation). Вычислительные процессы имеют *слабые* требования по предшествованию и отображаются несвязными графами. Это направление часто связывается с задачами балансирования и минимизации пересылок. При решении этих задач основой является определение— какой процесс (задание) будет выполняться на каком процессоре (ресурсе), а не определение порядка их выполнения [155]. При этом, в основном, решаются задачи пространственного распределения процессов (заданий). Задача — планирование (TaskScheduling). Вычислительные процессы имеют сильные требования по предшествованию и отображаются связными ациклическими графами (DAG). При этом, к решению задач первого направления добавляется также определение порядка выполнения заданий. На этом уровне часто решаются задачи минимизации суммарного времени выполнения полного DAG на выделенных или имеющихся ресурсах. Задачи по обоим направлениям, в общем случае, являются NP-полными или NP-сложными даже для двухпроцессорной системы [11] и имеют экспоненциальную временную сложность