

**1. Нащо впливає введення поняття «надійний стан системи»?**

Необходимо чтобы система была в надежном состоянии (ОС должна обеспечить всем текущим пользователям (процессам) завершение их заданий в течении конечного времени)

**2. Пропозиції Лебедева по підвищенню ефективності роботи ОС та як вони реалізовані в сучасних ОС.**

**3. Відмінність функцій програм системного введення / виведення від режиму спулінга.**

Режим буферизации для выравнивания скоростей при вводе и считывании информации из буфера. При работе программ системного ввода и/или режим спуллинга – согласование скоростей на входе и выходе.

**4.**

**Яка характеристика ОС покращується в змішаних дисциплінах обслуговування?**

Дисциплина обслуживания заявок – правило по которому решается задача обработки очереди заявок к каждому ресурсу. Бывают линейные (FiFo, Lifo, Random (нет одинаковых ресурсов)) циклические (RR, FBn, смешанная)

**5. Частиною якої програми є редактор зв'язку?**

Редактирование связей. (Выполнение связывания подпрограмм являющихся внешними по отношению к загружаемому модулю). (С помощью редактора связи мы получаем из объектного модуля загрузочный модуль имеющий всё для своего исполнения).

**6. Що таке супервізор і де він знаходиться?**

Режим супервизора — привилегированный режим работы процессора, как правило используемый для выполнения ядра операционной системы.

**7. Навіщо потрібні виконувати структурний аналіз вихідного графа?**

**8. Функції 3-го рівня багаторівневих систем програмування.**

Структурный анализ заявок и определение возможности распараллеливания (Задача распараллеливания)

**9.**

**Назвати рівні планування для багатопроцесорних ОС. Функції кожного.**

Семиуровневая модель – схема системы планирования с учетом (Масштабируемость, Разделяемость, Параллельность) 1. Предварительное (входное) планирование исходного потока заявок (задача фильтрации) 2. Структурный анализ взаимосвязи входного потока заявок по ресурсам с определением общих ресурсов (Анализ) 3. Структурный анализ заявок и определение возможности распараллеливания (Задача распараллеливания) 4. Адаптация распределения работ соответственно особенностям ВС (Задача адаптивирования) 5. Составление плана – расписания выполнения взаимосвязанных процедур. Оптимизация плана по времени решения и кол-ву ресурсов (Задача Оптимизации) 6. Планирование потока задач претендующих на захват времени процессора на каждый процессор – задача распределения. 7. Выделение процессорного времени, активизация задач. Перераспределение работ в ВС, при отказе оборудования (задача распределения – перераспределения).

**10. Коли застосовується запис до бібліотеки без каталогізації?**

Запис временных данных, которые удаляются после выполнения программы.

**11. Як визначити пріоритет процесу системного і проблемного?**

Добавляется планирование в пространстве; появляется проблема синхронизации процессов. ЕСЛИ в системе

нет одинаковых устройств то очереди обрабатываются по линейной схеме => планирование только во времени ИНАЧЕ планирование во времени и в пространстве.

## **12. Як змінюється організація обчислювального процесу для SMP і PC?**

### **13. Функція мети завдання динамічного планування.**

План составляется на том же оборудовании на котором выполняется решение задач во времени. Имеет ограничение на время составления плана. Динамическое планирование – задача планирования решается на том же оборудовании, что и выполняется план, который она составляет. Быстрое (чтобы не грузить оборудование) и неоптимальное решение Def. Джонсона – найти план распределения по ресурсам, при котором время решения не превышало бы критического с min кол-вом процов. Найти min кол-во ресурсов, чтобы задача решалась за min время. Для 2-х процов задача решается точно. Для одного в условиях RealTime.

### **14. Ідея побудови «базового» рішення.**

Базовое» решение оптимизируется по необходимому критерию - время выполнения, количество процессоров.

### **15. Сформулюйте принцип оптимальності Белмана.**

Эвристические (основанные на опыте людей), Пошаговое конструирование – разделение всего процесса решения на определенное кол-во шагов и нахождение оптимального или квазиоптимального решения. ЭТО – соответствует принципу оптимальности Белмана.