

Билет №17

1. Дати визначення непрямого доступу. Чи застосовується він зараз й коли?

Непрямой доступ – (косвенный доступ) – доступ к машине через управляющую программу (прообраз ОС, пакетный режим)

2. Що таке модульний принцип програмування?

Модульный принцип программирования – идея: программа разбита на куски. Каждый кусок выполняется с определенными соглашениями.

Свойства:

- каждый модуль обладает стандартностью внутр. Структуры
- взаимная независимость
- функц. Завершенность
- параметрическая универсальность

Виды модулей – вызывающий и вызываемый.

3. Сформулювати принцип оптимальності Белмана.

Принцип оптимальности Беллмана - замена решения исходной многомерной задачи последовательностью задач меньшей размерности.

4. Характеристика розподіленої операційної системи. Чим вона відрізняється від мережної?

Распределенная ОС - совокупность выч. узлов, связанных между собой каналами связи, с точки зрения usera представляют собой единое целое(одна вирт машина). Функции: расширяемость, масштабируемость, прозрачность
Сетевая ОС предоставляет пользователю некую виртуальную вычислительную систему, работать с которой гораздо проще, чем с реальной сетевой аппаратурой. В тоже время эта виртуальная система не полностью скрывает распределенную природу своего реального прототипа, то есть является виртуальной сетью. При работе юзер точно знает на какой машине выполняется задача.

5. Види завантажників та їх особливості. Коли який застосовується?

Загрузчик. Функции:

- Выделение памяти
- настройка
- связывание
- погружение

Виды загрузчиков: в зависимости от того, кто выполняет функции(прогр или система) разделяют:

- абсолютный загрузчик(выполняет только посл функцию)
- настраивающий (выполняет все функции (работает при создании .exe файла), компилятор ему должен сообщить доп. информацию: внешние и внутренние вектора переходов и биты переместимости (?) для настройки адресных констант)
- связывающий (непосредственно связывающий) – все функции выполняются по мере необходимости (ООП)

6. Які програми знаходяться у ядрі операційної системи? (Види програм)

Виды программ в ядре ОС: ядро супервизора, включающие только те программы, которые отвечают за реакцию системы.

Гугл: • обработка прерываний;

- создание и уничтожение процессов;
- переключение процессов из состояния в состояние;
- диспетчеризацию заданий, процессов и ресурсов;
- приостановка и активизация процессов;
- синхронизация процессов;
- организация взаимодействия между процессами;
- манипулирование PCB;
- поддержка операций ввода/вывода;
- поддержка распределения и перераспределения памяти;
- поддержка работы файловой системы;
- поддержка механизма вызова— возврата при обращении к процедурам;
- поддержка определенных функций по ведению учета работы машины;

7. Дати визначення програми простої структури. Коли використання такої структури неефективно?

Программа простой структуры: Программа которая при её исполнении не обращается к другим программам. Простейшая структура (имеют в своем теле все что нужно для выполнения)

8. Вхідна та вихідна інформація редактору зв'язків.

Входная и выходная информация редактора связей: вх- объектный модуль , вых – загрузочный модуль.

9. Як визначити вершини графу, що знаходяться на критичному шляху?

(?) это вершины, которые попадают на максимальный путь по графу (худшее время решения задачи)

10. Основна ідея «угорського алгоритму». Заміна якого блоку покращує його характеристики?

Венгерский алгоритм. Основная идея(википедия):

-если из всех элементов некоей строки или столбца вычесть одно и то же число , общая стоимость уменьшится на , а оптимальное решение не изменится;

-если есть решение нулевой стоимости, оно оптимально.

Алгоритм ищет значения, которые надо вычесть из всех элементов каждой строки и каждого столбца (разные для разных строк и столбцов), такие что все элементы матрицы останутся неотрицательными, но появится нулевое решение.

11. Навіщо потрібно водити характеристику «надійний стан системи»?

Надежное состояние системы - ОС может обеспечить всем процессам их выполнение в течение конечного времени.

12. Ідея алгоритмів, заснованих на стратегії критичного шляху.

Основная идея – кластеризация. Критический путь грузится на 1 проц и все пересылки зануляются.

13. Методи підвищення ефективності організації обчислювального процесу в ОС.

Методы повышения эффективности организации вычислительного процесса в ОС

Гугл:

1. Декомпозиция задачи.
2. Методы перебора состояний
3. Приведение целевой функции к сепарабельному виду.
4. Распараллеливание вычислительного процесса.
5. Разработка квазиоптимальных графиков.

14. Що таке шлях, що збільшує, шлях, що чергується?

Шлях, що збільшує, шлях, що чергується

(?) Теорема Герта: тах парасочетания получаются, если от свободной вершины нельзя найти увеличивающийся путь. Идея: для двудольного графа, случайно, выделяются ребра, которые не имеют инцендентных вершин.

Пространственный планировщик – задача поиск максимального парасочетания в двудольном графе. Вот там и используется увеличивающийся чередующийся путь.

15. Які задачі розв'язуються при статичному плануванні у паралельній ОС?

Стат планирование: NP-полные задачи, только для специализированных систем.

2 типа: минимальное время решения при заданном кол-ве проц; минимальное количество процессоров, чтобы время не превышало Tмах.