

БІЛЕТ №8

1 Дати визначення колективного доступу. Переваги й недоліки. Чи застосовується він зараз й де саме?

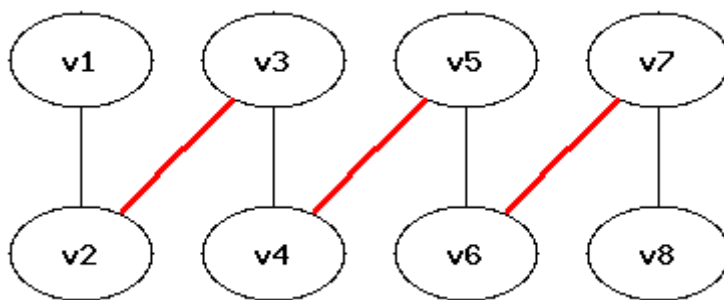
Коллективный доступ Предусматривающий доступ к ресурсам системы (система работает в многопрограммном режиме) многих пользователей, одновременная работа нескольких users на машине (Логика прерываний). Увеличивается скорость. Надежность машины. Доступ к ресурсам системы имеют несколько пользователей, система работает в многопрограммном режиме. У каждого пользователя свой терминал. Система должна быть очень надежной, а центральная машина - очень мощной.

2. Умова переходу процесу з активного стану в заблокований

Истечение кванта времени. (Выделенное время процессора для выполнения этого процесса завершилось и процессор занят другим процессом).

3. Чому усі алгоритми, засновані на теоремі Бержа про максимальне паросполучення непридатні для використання в динамічних планувальниках

Пусть зафиксировано некоторое паросочетание M . Тогда простая цепь $P = (v_1, v_2, \dots, v_k)$ называется чередующейся цепью, если в ней рёбра по очереди принадлежат - не принадлежат паросочетанию M . Чередующаяся цепь называется увеличивающей, если её первая и последняя вершины не принадлежат паросочетанию. Иными словами, простая цепь P является увеличивающей тогда и только тогда, когда вершина $v_1 \notin M$, ребро $(v_2, v_3) \in M$, ребро $(v_4, v_5) \in M$, ..., ребро $(v_{k-2}, v_{k-1}) \in M$, и вершина $v_k \notin M$.



Теорема Бержа (Claude Berge, 1957 г.). Паросочетание M является наибольшим тогда и только тогда, когда для него не существует увеличивающей цепи.

Доказательство необходимости. Пусть для паросочетания M существует увеличивающая цепь P . Покажем, как перейти к паросочетанию большей мощности. Выполним чередование паросочетания M вдоль этой цепи P , т.е. включим в паросочетание рёбра (v_1, v_2) , (v_3, v_4) , ..., (v_{k-1}, v_k) , и удалим из паросочетания рёбра (v_2, v_3) , (v_4, v_5) , ..., (v_{k-2}, v_{k-1}) . В результате, очевидно, будет получено корректное паросочетание, мощность которого будет на единицу выше, чем у паросочетания M (т.к. мы добавили $k/2$ рёбер, а удалили $k/2 - 1$ ребро).

Доказательство достаточности. Пусть для паросочетания M не существует увеличивающей цепи, докажем, что оно является наибольшим. Пусть \overline{M} — наибольшее паросочетание. Рассмотрим симметрическую разность $\overline{G} = M \oplus \overline{M}$ (т.е. множество рёбер, принадлежащих либо M , либо \overline{M} , но не обоим одновременно). Покажем, что \overline{G} содержит одинаковое число рёбер из M и \overline{M} (т.к. мы исключили из \overline{G} только общие для них рёбра, то отсюда будет следовать и $|M| = |\overline{M}|$). Заметим, что \overline{G} состоит только из простых цепей и циклов (т.к. иначе одной вершине были бы инцидентны сразу два ребра какого-либо паросочетания, что невозможно). Далее, циклы не могут иметь нечётную длину (по той же самой причине). Цепь в \overline{G} также не может иметь нечётную длину (иначе бы она являлась увеличивающей цепью для M , что противоречит условию, или для \overline{M} , что противоречит его максимальнойности). Наконец, в чётных циклах и цепях чётной длины в \overline{G} рёбра поочерёдно входят в M и \overline{M} , что и означает, что в \overline{G} входит одинаковое количество рёбер от M и \overline{M} . Как уже упоминалось выше, отсюда следует, что $|M| = |\overline{M}|$, т.е. M является наибольшим паросочетанием.

Теорема Берга даёт основу для алгоритма Эдмондса — поиск увеличивающих цепей и чередование вдоль них, пока увеличивающие цепи находятся.

4. Розподілена система — це набір незалежних комп'ютерів, що представляється їх користувачам єдиною об'єднаною системою.

У цьому визначенні обмовляються два моменти. Перший відноситься до апаратури: всі машини автономні. Другий стосується програмного забезпечення: користувачі думають, що мають справу з єдиною системою. Важливо обидва моменти. Можливо, замість того щоб розглядати визначення, розумніше буде зосередитися на важливих характеристиках розподілених систем. Перша з таких характеристик полягає в тому, що від користувачів приховані відмінності між комп'ютерами і способи зв'язку між ними. Те ж саме відноситься і до зовнішньої організації розподілених систем. Іншою важливою характеристикою розподілених систем є спосіб, за допомогою якого користувачі і додатки одночасно працюють в розподілених системах, незалежно від того, де і коли відбувається їх взаємодія.

Розподілені системи повинні також відносно легко піддаватися розширенню, або масштабуванню. Ця характеристика є прямим наслідком наявності незалежних комп'ютерів, але в той же час не указує, яким чином ці комп'ютери насправді об'єднуються в єдину систему. Розподілені системи зазвичай існують постійно, проте деякі їх частини можуть тимчасово виходити з ладу. Користувачі і додатки не повинні повідомлятися про те, що ці частини замінені або пошкоджені або що додані нові частини для підтримки додаткових користувачів або додатків.

Для того, щоб підтримати представлення різних комп'ютерів і мереж у вигляді єдиної системи, організація розподілених систем часто включає додатковий рівень програмного забезпечення, що знаходиться між верхнім рівнем, на якому знаходяться користувачі і додатки, і нижнім рівнем, що складається з операційних систем, як показано на рис. 1. Відповідно, така розподілена система зазвичай називається системою проміжного рівня (middleware).

5. Дать определение – транзитный модуль.

Программы, связанные с выполнением функций ОС, но не находящиеся постоянно в ОП называются транзитными. Эти программы вызываются в ОП по мере необходимости. – проги.,

которые могут понадобиться 4/ выполнения ф-ций ОС (но не резидентные) – могут вызываться по
оверлейной || динамически-последовательной схеме в транзитную зону.

6. Дать определение Табличный метод управления Метод когда решение управляющей системы

принимается на основании инфы, хранящейся в таблицах, содержащих данные состояния всех частей системы.

7. Цільова функція планування в реальному часі. Планирование в *реальном* времени характеризуется решением следующей задачи — определением плана решения совокупности задач с заданным временем исполнения и ограничениями по времени выходов задач из системы. Система планирования должна выполнять требование по минимизации суммарного времени отклонений реальных выходов задач из системы (выполнения сроков решений) от исходных временных ограничений (системе планирования сообщается время поступления задачи в систему, время решения и крайнее время выхода из системы) при полном соблюдении порядка следования работ.

8. Яка характеристика покращується у змішаних алгоритмах обслуговування й чому її треба покращувати?

Среднее время ожидания заявки. Для ускорения работы системы.

9. У чому відмінність між максимальним паросполученням й досконалим?

Найти максимальное паросочетание - значит найти максимальное число ребер графа в которых не совпадают координаты вершин, либо найти максимальное число единиц матрицы у которых не совпадают координаты.

10.Що таке критичні вершини? Вершина, которая является последней в графе, она определяет время решения.

11. . Чим визначається зміст програм, що знаходяться в оперативній пам'яті?

12. У чому складність для операційної системи в організації багатопрограмного режиму?

Организация защиты от взаимного влияния друг на друга на уровне оперативной и на уровне внешней памяти; разделение аппаратных и программных ресурсов; планирование (во времени, а в случае ПВС и в пространстве).

13.Как определить верхнюю границу кол-ва процессоров для погружения комплекса задач . Можно определить тах число процессоров, выше которого практически не выгодно иметь больше процессоров – тах ширина яруса.

**14.Классификация задач планирования(?) Динамическое Статическое
Балансовое В реальном времени**

15. минимальное к-во процессоров, когда время выполнения взаимосвязанных задач не превышает $T_{кр}$

$$N_{\min} = \left\lceil \frac{\sum_{i=1}^n t_i}{T_{кр}} \right\rceil$$