

Білет № 13

1. Який тип транслятора використовується при трансляції з традиційної машинної мови у команди машини та як він реалізується?

По идее, Ассемблер. Процесс трансляции программы на языке ассемблера в объектный код принято называть ассемблированием. В отличие от компилирования, ассемблирование — более или менее однозначный и обратимый процесс. В языке ассемблера каждой мнемонике соответствует одна машинная инструкция, в то время как в языках программирования высокого уровня за каждым выражением может скрываться большое количество различных инструкций. В принципе, это деление достаточно условно, поэтому иногда трансляцию ассемблерных программ также называют компиляцией.

2. Визначення метаобчислень (суперобчислень)

Теория метасистемных переходов - это теория эволюции. Условия реализации эволюции можно создать в мире программ. Достаточно написать для них метапрограммы, то есть программы, преобразующие другие программы. Последовательное и (формально) бесконечное применение метапрограмм к программам и к самим себе будет порождать новые программы. При этом могут возникать метасистемные переходы, приводящие к качественно новым возможностям программ. Созданием метапрограмм (новых метасистем) над программами на основе методов анализа и преобразования программ и занимается теория метавычислений. =))

3. Чому в розподілених системах переважніше використовувати крупнозернисте планування?

Крупнозернистое планирование – когда время выполнения узла больше времени пересылки

4. Дії, що виконує програма ініціалізації ядра. Звідки вона викликається?

При инициализации ядра и системы активизируются и исполняются следующие процессы операционной системы: администратор памяти, программа работы с ВУ, программа поддержки файловой системы и процесс, подчиненный таймеру. Вызывается BIOS=>первичный загрузчик=>вторичный загрузчик=>прогр. Иниц.ядра

5. Функції завантажника, що налаштовує. Яку інформацію повідомляє йому компілятор?

Распределение памяти, настройка, редактирование, загрузка. Инфу для него готовит компилятор. Настраивающий – работает в загр. Модуле которому больше ничего не надо. Постепенно его ф-ции взял на себя редактор связей. Непосредственно в настраивающем загрузчике каждый модуль может транслироваться отдельно. Чтобы передать сообщение редактору связей надо ему непосредственно указать, что надо транслировать. В каждом модуле в начале трансляции выделяются вектора перехода, внешние и внутренние. (Экспорт и Импорт процедур и ф-ций). Кроме того для выполнения настройки каждая команда отмечается битом переместимости. ОС выделяет и пользуется глобально выделенной памятью, а загрузчик с локальной.

6. Сформулювати теорему про обов'язкові призначення.

С помощью анализа матрицы выделяют назначение, которое есть обязательным еще до нахождения базового решения. Теорема 1 :

Если в двудольном графе существует вершина, которой инцидентно только одно ребро, то вершины, инцидентные этому ребру, должны быть взяты в решение

7. Як визначити об'єм пам'яті, необхідний для завантаження програми оверлейної структури?

Корневой сегмент + самый большой модуль.

8. Основна особливість пріоритетних дисциплін обслуговування. Види.

Приоритетное обслуживание – заявка на вход системы с заданным приоритетом. Относительный приоритет – не может прервать задачу на ресурсе, даже если она имеет низкий приоритет. Абсолютный приоритет – прерывает задачу на ресурсе если та имеет более низкий приоритет. Динамический приоритет – ЕСЛИ возникает опасность бесконечного откладывания. ЕСЛИ по мнению sys, заявка слишком долго занимает ресурс, ТО ее приоритет понижается. ЕСЛИ заявка очень долго ожидает ресурса ТО ее приоритет

повышается.(QoS). Если заявка с более высоким приоритетом не прерывает обслуживание заявки с низким приоритетом – относительная ДО без вытеснения, соответственно, если наоборот - абсолютная ДО с вытеснением.

Приоритеты бывают:

- статические (сразу задаются заявке и не изменяются в процессе)
- динамические (приоритеты меняются в зависимости от $t_{\text{ожид.}}$ или $t_{\text{обслуж.}}$).

9. Дати визначення паралельної системи.

Параллельные вычислительные системы — это физические компьютерные, а также программные системы, реализующие тем или иным способом параллельную обработку данных на многих вычислительных узлах.[1]
Вики

10. Ідея розв'язання задачі планування методом «пізнього» планування.

придерживаем задачу без увеличения критического пути

11. Задача «наповнення рюкзака». Для розв'язання якої задачі планування можна її застосовувати?

12. Яка інформація міститься у MBR?

код загрузчика, таблицу разделов (partition table) и специальную сигнатуру

13. Дати визначення процесу. Коли це поняття стало застосовуватися й чому?

Процесс – это траектория процессора в адресном пространстве машины

14. Застосування GRID систем.

Эта технология применяется для решения научных, математических задач, требующих значительных вычислительных ресурсов. Грид-вычисления используются также в коммерческой инфраструктуре для решения таких трудоёмких задач, как экономическое прогнозирование, сейсмоанализ, разработка и изучение свойств новых лекарств.+адронный коллайдер

15. Відмінність завдання від задачі. Де зберігається завдання?

Задание – внешняя единица работы системы (описывается на специальном языке). Задание загружается в систему только тогда , когда система имеет свободные ресурсы и преобразуется в задачу.

Задача – внутренняя единица работы системы,для которой система выделила ресурсы кроме процессорного времени.Задача фиксируется в системе если ей выделены системные ресурсы (блок управления процессом).