

1 **Дать определение. Системное программное обеспечение**

Совокупность программ и их описание.

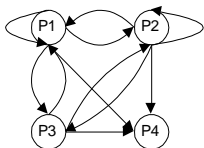
2 **Классификация ОС.**

Однопрограммные, многопрограммные, однопроцессорные, многопроцессорные, распределённые, виртуальные.

3 **Условие создания процесса.**

Корректность описания процесса на языке описания процессов и выделение ресурсов

4 **Показать все связи состояний P1-P4.**



5 **Виды загрузчиков. Их основные отличия.**

Отличаются выполнением основных 4-х функций: распределения памяти, настройки, редактирования и загрузки. Загрузчики делятся: Абсолютный загрузчик, настраивающий загрузчик, непосредственно - связывающий загрузчик.

6 **Дать определение. Генерация системы.**

Процесс отбора из дистрибутива тех программных модулей, которые будут использоваться - т.е. создание операционной среды для конкретной конфигурации вычислительной системы.

7 **Дать определение. Программа простой структуры.**

Программа которая при её исполнении не обращается к другим программам.

8 **Функции редактора связей.**

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).

9 **Что такое система обслуживания прерываний.**

Комплекс программно-аппаратных средств, обеспечивающих приостановку активного процесса и переход к обработке прерывания.

10 **Обработка программных прерываний.**

11 **Что такое просмотр команд вперед и какая характеристика улучшается.**

Предварительная выборка (увеличивается быстродействие)

12 **Что такое «свопинг».**

Свопинг - способ реализации многопрограммного режима работы на однопроцессорной машине.

13 **Способы реализации многопрограммных режимов работы.**

Свопинг, разделение времени, классическое мультипрограммирование, параллельная обработка.

14 **Дать определение тупика.**

Тупик - это ситуация, из которой система не может выйти; ситуация, когда процесс ждёт события, которое никогда не произойдёт

15 **Виды планирования и их особенности.**

Динамическое (план составляется на том же оборудовании на котором выполняется решение задач во времени); статическое (план решения задач может составляться на другом оборудовании); балансовое (решение задачи балансирования узлов в сети).

16 **Перечислить способы реализации многопрограммного режима работы.**

Свопинг, разделение времени, классическое мультипрограммирование, параллельная обработка.

17 **Условие перехода процесса из активного состояния в готовое**

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

18 **Свойства модуля.**

Стандартность внутренней структуры, функциональная завершенность, параметрическая универсальность, взаимная независимость.

19 **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.

20 **Какие программы находятся в ядре ОС. (Виды программ)**

- 21 **Какую информацию и как компилятор передает настраивающему загрузчику**  
Сообщают загрузчику о том что в данной команде есть адресная константа с помощью битов переместимости.
- 22 **Что такое система прерывания**
- 23 **Почему и когда блокируется система прерывания**  
При дешифрации текущего прерывания
- 24 **Что такое глубина системы прерываний и от чего зависит эта характеристика**  
- степень вложенности прерываний. (определяется количеством старых PSW которые можно поместить в постоянную область памяти или в стеке.)
- 25 **Что такое «спулинг».**  
Режим буферизации для выравнивания скоростей при вводе и считывании информации из буфера.
- 26 **Что такое псевдонумерация прерываний.**
- 27 **Особенности ОС локальных ВС**
- 28 **Структура модуля**  
Заголовок, инициализирующая часть и тело (+-)
- 29 **Функции загрузчика.**  
Распределение памяти, настройка, редактирование, загрузка.
- 30 **Основные действия программы начальной загрузки. Где она хранится.**  
В MBR. Активизация программ начальной загрузки.
- 31 **Как определить объем памяти необходимой для загрузки программы динамическо - последовательной структуры**
- 32 **Какая характеристика улучшается в смешанных алгоритмах обслуживания.**
- 33 **Зачем используется CSW.**  
CSW - слово состояния канала.
- 34 **Объяснить - косвенный ввод-вывод**
- 35 **Дать определение. Математическое обеспечение**  
- это совокупность средств и методов, необходимых для решения поставленной задачи
- 36 **Особенности распределенных ОС.**
- 37 **Назначение частей ОС ориентированной на пользователя и на hardware.**  
Часть системы, ориентированной на «hardware» скрывает от пользователя те физические особенности системы, которые ему не нужны. (обеспечение взаимодействия устройств, процессов и т.д.). Часть ОС, ориентированной на пользователя, обеспечивает интерфейс с пользователем.
- 38 **Функции абсолютного загрузчика.**  
Распределение памяти, настройка, редактирование;
- 39 **Как определить объем памяти для загрузки программы простой структуры.**  
Необходимо знать размер самой программы + корневой каталог.
- 40 **Почему алгоритм «Корбата» лучше 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.)
- 41 **Где находятся новые PSW.**  
Значение нового PSW находится в векторе прерывания с номером n, где n указывает номер прерывания обработчик которого нужно взять.
- 42 **Где находятся новые PSW. Модификация в различных ОС.**  
41+B векторе прерывания.
- 43 **Что такое закливание приоритетов**
- 44 **Функции библиотекаря.**  
Работа с библиотекой : поиск, запись, удаление, редактирование, копирование, запись без каталогизации.
- 45 **Основные отличия компилятора от интерпретатора.**  
Интерпретатор выполняет перевод части проги в машинные команды и тут же ее выполняет, выполняя все необходимые настройки адресных констант. Компилятор создает объектный модуль, который затем

обрабатывается редактором связей. Во время работы интерпретатора исполняемый код программы записывается в фиксированное место и управление передается на стартовый адрес программы.

46 **Дать определение. Многопрограммный режим работы.**

Режим, когда в системе находятся несколько задач в разной стадии исполнения, и каждая из них может быть прервана другой.

47 **Основная особенность ОС Unix.**

ОС взаимодействует с аппаратурой непосредственно, обеспечивая обслуживание программ и их независимость от деталей аппаратной конфигурации.

48 **Дать определение. Область сохранения.**

Область сохранения - область в которую записывается информация о прерванном процессе, и откуда система берет данные при восстановлении процесса.

49 **Функции настраивающего загрузчика.**

Распределение памяти, настройка, редактирование, загрузка. Инфу для него готовит компилятор.

50 **Как определить объем памяти необходимой для загрузки программы оверлейной структуры.**

Корневой сегмент + самый большой модуль.

51 **Основная особенность приоритетных дисциплин обслуживания**

Заявки имеют приоритет.

+1 ))

52 **Зачем используется CAW**

53 **Что такое насыщение системы прерываний**

54 **Что такое временная библиотека.**

Библиотека записанная без каталогизации.

55 **Дать определение. Режим разделения времени**

Реж. раз. Времени - режим совмещающий мультипрограммирование и параллельную обработку, плюс возможность привелигированным пользователям иметь прямой доступ к ресурсам системы.

56 **Дать определение. Дисциплины обслуживания заявок. Их классификация.**

Дисциплина обслуживания заявок - это совокупность правил по которой обслуживаются заявки. Бывают линейные (FiFo,Lifo,Random) циклические (RR,FBn,смешанная)

57 **Какая программа выполняет запись информации в область сохранения и какая информация там записывается**

58 **Какой тип загрузчика работает с модулями с расширением COM и EXE.**

С COM - абсолютный; с EXE - настраивающий.

59 **Где находится программа начальной загрузки. Ее функции.**

В MBR. Активизация программ начальной загрузки.

60 **Как определить объем памяти необходимой для загрузки программы динамическо - параллельной структуры.**

Корневой сегмент + наибольший суммарный объем модулей, которые могут выполняться параллельно.

61 **Определение прерывания**

-событие, требующее от системы прекращение работы активного процесса и перехода к обработке другого задания.

62 **Структура библиотеки.**

Сама библиотека, управляющая программа, каталог.

63 **Что такое виртуальная ОС.**

Так называемая ОС, которая позволяет многим пользователям работающим на одной и той же технической базе одновременно работать в различных операционных средах.

64 **Дать определение. Классическое мультипрограммирование.**

Режим истинного совмещения, когда параллельно исполняемые задачи занимают различное оборудование.

65 **Дать определение. Процесс. Состояния процесса.**

Это любая выполняемая работа в системе; это динамический объект (внутренняя единица работы) системы, которому она выделяет ресурсы; траектория процессора в адресном пространстве.

66 **Связь модулей по данным. Виды связей.**

Виды: по данным и по управлению. По данным может быть через общие регистры или через адрес списка параметров.

- 67 **Дать определение. Резидентный модуль**  
-модуль, который резидентно находится в памяти.
- 68 **Дать определение. Табличный метод управления.**  
Метод когда решение управляющей системы принимается на основании инфы, хранящейся в таблицах, содержащих данные состояния всех частей системы.
- 69 **Какая системная программа подготавливает информацию для работы загрузчика.**  
Компилятор.
- 70 Где фиксируется прерывание ввода-вывода.
- 71 **Что такое запись в библиотеку без каталогизации.**  
Запись временных данных, которые удаляются после выполнения программы.
- 72 **Перечислить классы прерываний по приоритету.**  
От схем контроля машиной; внешние; прерывания по вводу/выводу; по обращению к супервизору; программные/
- 73 **Дать определение. Параллельная обработка.**  
-Режим кажущегося совмещения (с последующим крещением), квантования.
- 74 **Условие перехода процесса из подготовленного состояния в готовое.**  
Выделение ресурсов.
- 75 **Чем отличается традиционный машинный язык программирования от команд машины.**  
Команды машины - это самые элементарные выполняемые команды (самый низкий уровень); традиционный машинный язык - это набор процедур из элементарных команд, выполняемых на самом низком уровне.
- 76 **Почему применяются многоуровневые системы программирования.**  
Для облегчения разработки компилятора (ставится несколько уровней трансляторов).
- 77 **Классы прерывания.**  
От схем контроля машины, внешние, по вводу/выводу, по обращению к супервизору, программные.
- 78 **Кто вырабатывает сигнал прерывания по вводу-выводу.**  
Аппаратура В/В при изменении состояния каналов В/В, а также при завершении операций В/В.
- 79 Особенности стирания программы из библиотеки на внешнем носителе
- 80 **Какая системная программа подготавливает команду начать ввод-вывод.**  
Обработчик прерываний.
- 81 **Дать определение. Косвенный доступ.**  
Доступ осуществляющийся через управляющую программу; предусматривает общение с вычислительной установкой косвенно.
- 82 **Условие перехода процесса из готового состояния в активное.**  
Выделение кванта времени процессора, запуск.
- 83 **Виды загрузчиков.**  
Абсолютный, настраивающий, непосредственно - связывающий.
- 84 **Входная и выходная информация РЕДАКТОРА СВЯЗЕЙ.**  
Входная - объектный модуль, выходная - загрузочный модуль.
- 85 **Виды PSW где они хранятся.**  
Текущий (в регистрах процессора), новый (в векторах прерываний), старый (в постоянной области памяти вместе с векторами прерываний; в стеке той программы которая прерывается или в PSW прерванного процесса).
- 86 **Кто вырабатывает сигнал прерывания по обращению к супервизору.**  
Выполняемый процесс.
- 87 **В чем разница между «утилитами» и «сервисными программами».**  
Нет никакой разницы. Утилита это и есть сервисная программа.(другой вариант: утилиты входят в состав сервисных программ).
- 88 **Какая часть ОС обрабатывает сигналы прерываний.**  
Ядро.
- 89 **Дать определение. Работа в реальном времени.**  
Если время реакции системы детерминировано, определено внешними факторами.
- 90 **Условие перехода процесса из активного состояния в заблокированное.**  
Ожидание события, освобождения ресурса, конца операции ввода/вывода.

91 **Отличие загрузочного модуля от объектного. Участие системных программ в их создании.**

Загрузочный модуль имеет все для своего выполнения. Написан на машинно-ориентированном языке, но быть выполненным не может. При использовании компилятора из исходного модуля получается объектный модуль, при использовании редактора связи из объектного получается загрузочный, далее из него при использовании загрузчика получается исполняемый модуль.

92 **Дать определение. Программа оверлейной структуры.**

-Задача, разделяемая на модули. Модули находящиеся на одном оверлейном уровне не могут одновременно находиться в ОП.

93 **Чем алгоритм FiFo лучше LiFo.**

Дисциплина FiFo минимизирует дисперсию ожидания.

94 **Информация хранящаяся в PSW.**

Уникальный идентификатор процесса (имя); текущее состояние процесса; приоритет процесса; указатели участка памяти выделенного программе, подчиненной данному процессу; указатели выделенных ему ресурсов; область сохранения регистров; права процесса (список разрешенных операций); связи зависимости в иерархии процессов (список дочерних процессов, имя родительского процесса); пусковой адрес проги, подчиненной данному процессу. А также дополнительная инфа для синхронизации процессов, 4 поля: 1-2: поля для организации цепочки связи; 3-4: поля для организации цепочки ожидания.

95 **От чего зависит момент прерывания.**

96 **Дать определение. Обработывающие программы ОС.**

Программы выполнения стандартных (в рамках ОС) функций, обработки исключительных ситуаций.

97 **В чем сложность для ОС в организации многопрограммного режима работы.**

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).

98 **Условие перехода процесса из заблокированного состояния в готовое.**

Наступление события, освобождения ресурса, завершение ввода/вывода.

99 **Особенность создания процесса при рекурсивном обращении**

100 **Отличие загрузочного модуля от абсолютного.**

Загрузочный не может быть выполненным (абсолютный модуль определен в адресном пространстве).

101 **Дать определение. Резидентный том. Резиденция системы.**

Место где находится резиденция системы называется резидентным томом, а сама система резиденцией.

102 **Дать определение. Программа динамическо - последовательной структуры. Чем она лучше других систем.**

Программы настроенные с помощью модульного принципа, представляется в перемещаемом виде, могут подгружаться по мере необходимости с организацией связей по управлению и данными.

103 **Почему алгоритм RR лучше FiFo.**

Длинная заявка не может захватить ресурс т.к., вводится понятие квант. Если заявка в RR не обслужена полностью, то по истечении кванта она помещается в очередь; в FiFo заявка будет обслуживаться до ее выполнения.

104 **Смысл смены PSW. Чем ограничена глубина прерывания.**

PSW служит для реализации переключения процессов. В случае сохранения старых PSW в постоянной области памяти глубина определяется классов прерываний. Если старые PSW сохраняются в стеке то глубина определяется размером памяти отведенной под стек.

105 **Из чего складывается время ожидания при обработки сигнала прерывания.**

Время на сохранения состояния прерываемой задачи; время на сохранения адреса возврата; загрузка вектора прерываний.

106 **Дать определение. Управляющие программы ОС.**

Программы постоянно находящиеся в памяти (резидентные) организующие корректное выполнение процессов и функционирование всех устройств системы при решении задач. Составляют ядро ОС.

107 **Что такое прямой ввод-вывод.**

Когда не используются управляющие программы.

108 **Виды модулей. Участие системных программ в преобразовании модулей**

Виды: 1) исходный (текст программы); 2) объектный; 3) загрузочный; 4) исполняемый (абсолютный). Системные программы (компилятор, редактор связей, загрузчик) используются для преобразования от 1) к 4).

#### 109 **Трудность планирования параллельных процессов**

Добавляется планирование в пространстве; появляется проблема синхронизации процессов.

#### 110 **Условие возникновения тупика.**

1) Условие взаимного исключения; 2) Условие ожидания; 3) Условие перераспределения; 4) Круговая цепь ожидания.

#### 111 **Дать определение. Резиденция системы.**

Вариант ОС полученный после генерации (установки)

#### 112 **Дать определение. Программа динамическо - параллельной структуры**

Динамически - последовательная структура с указанием о том какие модули могут выполняться параллельно

#### 113 **Почему алгоритм FBN лучше RR**

Заявки в последней очереди выполняются без прерывания обслуживания => уменьшение временных затрат обусловленных прерыванием

#### 114 **Где находятся старые PSW**

Старые PSW могут храниться в постоянной области памяти вместе с векторами прерываний; в стеке той программы которая прерывается или в PCB прерванного процесса.

#### 115 **Что такое маскирование прерываний. Как оно осуществляется**

После того как прерывание замаскировано контроллер игнорирует появление сигнала этого прерывания. Осуществляется путем загрузки регистра маски КПП.

#### 116 **Связь модулей по управлению. Какие операции выполняются и какими программами**

Связь по управлению предусматривает сохранение состояния модуля на момент перехода на вызываемый модуль, причем действия по этому сохранению выполняет вызываемый модуль. Операции: возврат из  $i$  модуля только в  $i-1$ ; возврат из  $i+1$  в  $i$ . Виды программ могут быть: повторно не исполняемые, повторно исполняемые, чистые процедуры.

#### 117 **Что такое насыщение системы прерываний**

#### 118 **Почему и когда блокируется система прерывания**

#### 119 **Виды модулей.**

1) исходный (текст программы); 2) объектный; 3) загрузочный; 4) исполняемый (абсолютный).

#### 120 **Дать определение. Коллективный доступ.**

Предусматривающий доступ к ресурсам системы (система работает в многопрограммном режиме) многих пользователей.

#### 121 **Дать определение и особенности динамического планирования**

План составляется на том же оборудовании на котором выполняется решение задач во времени. Имеет ограничение на время составления плана.

#### 122 **Стратегии борьбы с тупиками.**

Предотвращение тупика, обход тупика, обнаружение тупика, восстановление после тупиков.

#### 123 **В чем разница между частично упорядоченным и полно упорядоченными системами прерываний**

В частично - упорядоченной системе в отличие от простой схемы полно - упорядоченной системы вводится двухуровневая система приоритетов. 1 уровень - среди классов, 2 уровень - внутри каждого класса, при этом класс и само прерывание определяется по полно упорядоченной схеме. В полно - упорядоченной ускоряется поиск, а в частично - упорядоченной упрощаются процедуры поиска.

#### 124 **Дать определение. Область сохранения. Чем определяется размер**

Область сохранения - область в которую записывается информация о прерванном процессе, и откуда система берет данные при восстановлении процесса. Размер области сохранения зависит от того количества данных о процессе, которые необходимы системе для возобновления процесса.

#### 125 **Особенности программного обеспечения распределенных систем обработки информации**

Отсутствие общей памяти приводит к невозможности определения общего состояния с помощью множества совместных переменных, а невозможность совместного обращения к памяти и различие в задержках передач сообщений приводит к тому что при определении состояния какого либо элемента системы из двух различных точек можно получить разные результаты. Выполнение работы

распределяется в узлах исходя из соображения пропускной способности всей системы. Распределенные системы имеют высокий уровень организации параллельных вычислений.

126 **Отличие загрузочного модуля от абсолютного. Какие системные программы участвуют в их создании.**

Загрузочный выполняемый быть не может.. Компилятор, редактор связей, загрузчик.

127 **Дать определение. Инициализация системы. Какая программа выполняет эти действия**

-процесс создания ядра системы который включает не только перезапись программ, но и системных структур, обеспечивающих знания системы о ее параметрах. Программа - загрузчик.

128 **Характеристика задач планирования с предшествованием**

129 **Дать определение - двудольный граф.**

Граф имеющий только два уровня, связь между вершинами одного уровня невозможна.

130 **Чем отличается загрузка EXE и COM модулей.**

Для \*.exe требуется настройка адресных констант.

131 **Дать определение. Инсталляция.**

Процесс локальной настройки ОС по желанию пользователя.( Если при изменении параметров системы требуется установка с дистрибутива.)

132 **Как определить объем памяти необходимой для загрузки программ простой структуры и оверлейной.**

Простой - размер самой программы + корневой каталог. Оверлейной - Корневой сегмент + самый большой модуль.

133 **Чем ограничена глубина прерывания.**

определяется количеством старых PSW которые можно поместить в постоянную область памяти или в стеке

134 **Задачи решаемые при обходе тупиков.**

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

135 **Объяснить - поиск максимального паросочетания.**

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.

136 **Условия перехода из P3 в P1 или P2.**

Выход из прерывания, вызов сист. Функции из прерывания

137 **Чем отличается матрица связности от матрицы инцидентности**

138 **Как определить минимальное к-во процессоров, необходимых для загрузки параллельной задачи**

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

139 **Дать определение. Косвенный доступ**

140 **Задачи решаемые при статическом планировании**

Поиск min-го кол-ва процессоров, необходимых для решения комплекса информационно- и по управлению взаимосвязанных задач за время, не превышающее заданное или критическое; поиск плана решения заданного комплекса информационно- и по управлению взаимосвязанных задач на заданном кол-ве процессоров за минимальное время.

141 **Функции настраивающего загрузчика. Какая программа готовит для него информацию**

Распределение памяти, настройка, редактирование, загрузка. Информацию для него готовит компилятор.

142 **Сущность алгоритма банкира.**

Обход тупиков.

143 **Способы описания графа**

Табличный, графический, аналитический и словесный.