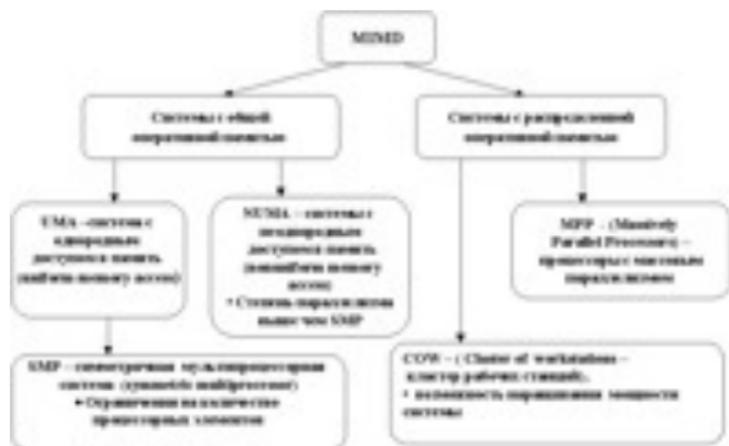


- 1) **Первое поколение компьютеров** - электронно-вакуумные лампы, 1940-1950. Зарождение класса сервисных, управляющих программ. Зарождение языков программирования. Однопользовательский, персональный режим.
- 2) **Второе поколение компьютеров** - полупроводниковые приборы: диоды и транзисторы, 1950-1960, пакетная обработка заданий, мультипрограммирование, языки управления заданиями, файловые системы, виртуальные устройства, операционные системы.
- 3) **Третье поколение компьютеров** - интегральные схемы малой и средней интеграции, 1960-1970, аппаратная унификация узлов и устройств, создание семейств компьютеров, унификация компонентов программного обеспечения.
- 4) **Компьютеры четвертого поколения** - большие и сверхбольшие интегральные схемы, 1970-х – настоящее время. «Дружелюбность» пользовательских интерфейсов, сетевые технологии, безопасность хранения и передачи данных.
- 5) **Вычислительная система** - совокупность аппаратных и программных средств, функционирующих в единой системе и предназначенных для решения задач определенного класса.
- 6) **Уровни вычислительной системы** снизу-вверх: аппаратный, уровень физических устройств, уровень логических устройств, системы программирования, прикладные системы.
- 7) **Драйвер физического устройства** – программа, основанная на использовании команд управления конкретным физическим устройством и предназначенная для организации работы с данным устройством.
- 8) **Логическое/виртуальное устройство (ресурс)** – устройство/ресурс, некоторые эксплуатационные характеристики которого (возможно все) реализованы программным образом.
- 9) **Драйвер логического/виртуального ресурса** - программа, обеспечивающая существование и использование соответствующего ресурса.
- 10) **Ресурсы вычислительной системы** - совокупность всех физических и виртуальных ресурсов.
- 11) **Операционная система** - это комплекс программ, обеспечивающий управление ресурсами вычислительной системы.
- 12) **Система программирование** – это комплекс программ, обеспечивающий поддержание жизненного цикла программы в вычислительной системе.
- 13) **Этапы жизни программы:** проектирование, кодирование, тестирование, отладка, изготовление.
- 14) **Проектирование** - исследование задачи, исследование характеристик объектной среды (как объектная среда будет связана с нашей системой).
- 15) **Объектная среда** – это та ВС, в рамках которой продукт будет функционировать.
- 16) **Инструментальная среда** – это ВС, которая будет использована для разработки программ.
- 17) **Построение кода на основании спецификаций при использовании языков программирования, трансляторов, средств для использования библиотек и средств для разработки программных продуктов.** Результатом этапа кодирования являются исполняемые модули, объектные модули, исходные тексты программ и библиотеки.
- 18) **Тестирование** – это проверка спецификаций функционирования программы на некоторых наборах входных данных.
- 19) **Отладка** – процесс поиска, анализа и исправления зафиксированных при тестировании и эксплуатации ошибок.
- 20) **Современные разработки программного обеспечения:** каскадная модель, каскадно-итерационная, спиральная.
- 21) **Прототип** - программа, частично реализующая функциональность и внешний интерфейс разрабатываемой системы.

- 22) **Виртуальная машина** - программно реализованные аппаратные средства. Спецификация некоторой вычислительной среды
- 23) **Прикладная система** – программная система, ориентированная на решение или автоматизацию решения задач из конкретной предметной области.
- 24) **Принципы работы компьютера Фон Неймана:** принцип двоичного кодирования, принцип хранимой программы, принцип программного управления.
- 25) **Центральный процессор (ЦП)** – компонент компьютера, обеспечивающий выполнение программ. Программы, выполняемые в рамках процессора, координируют работу ОЗУ и внешних устройств.
- 26) **Оперативно запоминающее устройство (ОЗУ)** – устройство хранения данных, в котором размещается исполняемая в данный момент программа и из которого выбираются команды и данные этой программы.
- 27) **Внешние устройства** – программно управляемые устройства, входящие в состав компьютера.
- 28) **Арифметически-логическое устройство (АЛУ)** - устройство, реализующее команды, которые подразумевают обработку данных.
- 29) **Устройство управления (УУ)** - обеспечивает последовательный выбор команд, которые необходимо выполнить программе, их контроль, дешифровку и, в зависимости от типа команды, последующую обработку.
- 30) **ОЗУ состоит** из ячеек памяти, которые состоят из **тегов** (поле служебной информации) и **машинных слов** (поле программно изменяемой информации).
- 31) **Производительность оперативной памяти** - скорость доступа процессора к данным, размещенным в ОЗУ.
- 32) **Время доступа (access time- taccess)** - время между запросом на чтение слова из оперативной памяти и получением содержимого этого слова.
- 33) **Длительность цикла памяти (cycle time - tcycle)** - минимальное время между началом текущего и последующего обращения к памяти. (tcycle>taccess)
- 34) **Расслоение ОЗУ** – один из аппаратных путей решения проблемы дисбаланса в скорости доступа к данным, размещенным в ОЗУ и производительностью ЦП.
- 35) **Регистровая память** – совокупность устройств памяти ЦП ,предназначенных для временного хранения операндов, информации, результатов операций. (Регистр Адреса (РА), Регистр Результата (РР),Слово – состояние процессора (ССП или PSW), Регистры внешних устройств (РВУ) , Регистр указатель стека (PVC).
- 36) **КЭШ память** - (Аппаратное решение) буферизация работы процессора с оперативной памятью.
- 37) **Стратегии вытеснения КЭШ памяти** — случайное, наименее популярного.
- 38) **Процесс вытеснения** — сквозное, с обратной связью.
- 39) **Прерывание** - событие в компьютере, при возникновении которого в процессоре происходит predetermined последовательность действий.
- 40) **Внутреннее прерывание** - инициируются схемами контроля работы процессора.
- 41) **Внешнее прерывание** - события, возникающие в компьютере в результате взаимодействия.
- 42) **Внешние запоминающие устройства (ВЗУ)**– устройства, предназначенные для хранения данных и программ.
- 43) **Устройства ввода и отображения информации** – осуществляют ввод из вне некоторой информации и отображение ее в виде некоторых результирующих данных.
- 44) **Устройства приема и передачи данных** используются для получения данных с других компьютеров, «из вне».

- 45) **ВЗУ Последовательного доступа:** Магнитная лента
- 46) **ВЗУ Прямого доступа:** магнитные диски, магнитный барабан, магнито-электронные ВЗУ прямого доступа.
- 47) **Поток информации** - программно заданная информация о том, что необходимо прочесть или записать или переместить данные из одного места в другое.
- 48) **Поток данных** - непосредственно ответ на управляющее воздействие и связанное с этим ответом перемещение данных от внешнего устройства в ОП или в ЦП.
- 49) **DMA (direct memory access)** – устройство, позволяющее обращаться к ОЗУ из ВЗУ без посредника.
- 50) **Иерархия памяти:** регистры общего назначения, КЭШ 1-го уровня, КЭШ 2-го уровня, ОЗУ, внешнего запоминающего устройство с внутренней КЭШ-буферизацией, внешнего запоминающего устройство без внутренней КЭШ-буферизации, внешнего запоминающего устройство длительного хранения.
- 51) **Мультипрограммный режим** - режим при котором возможна организация переключения выполнения с одной программы на другую.
- 52) **Аппаратные средства компьютера, необходимые для поддержания мультипрограммного режима:** аппарат прерываний (хотя бы прерывание по таймеру), аппарат защиты памяти, специальный режим ОС (привилегированный).
- 53) **Регистровые окна** – один из способов решения проблемы вложенных процедур.
- 54) **Аппарат виртуальной памяти** – аппаратные средства компьютера, обеспечивающие преобразование (установление соответствия) программных адресов, используемых в программе адресам физической памяти в которой размещена программа при выполнении.
- 55) **Базирование адресов** – реализация одной из моделей аппарата виртуальной памяти. При базировании выделяется регистр, в котором будет храниться адрес, начиная с которого размещается программа.
- 56) **Страничная организация** — аппаратная организация памяти, при которой все пространство делится на фрагменты одного размера (обычно 2ⁿК)
- 57) **Виртуальное адресное пространство** – множество виртуальных страниц, доступных для использования в программе. Количество виртуальных страниц определяется размером поля «номер виртуальной страницы» в адресе.
- 58) **Физическое адресное пространство** – оперативная память, подключенная к данному компьютеру. Физическая память может иметь произвольный размер (число физических страниц может быть меньше, больше или равно числу виртуальных страниц).
- 59) **Классификация Флинна:** поток управляющей информации – собственно команд (инструкций), и поток данных. Считаем потоки данных и команд независимыми (условно). Рассматриваем все возможные комбинации.
- 60) **ОКОД (SISD – single instruction (одиночный поток команд), single data stream, (одиночный поток данных))** Традиционные компьютеры, которые мы называем однопроцессорными.
- 61) **ОКМД(SIMD – single instruction(одиночный поток команд), multiple data stream(множественный поток данных))** Для каждой команды порция данных (векторная или матричная обработка данных)
- 62) **МКОД (MISD – multiple instruction(множественный поток команд), single data stream(одиночный поток данных))** – это вырожденная категория, считается, что ее нет. (Но есть вырожденные случаи — обработка графики)
- 63) **МКМД (MIMD - multiple instruction(множественный поток команд), multiple data stream(множественный поток данных))**

Хм, ну это тут лучше всего будет смотреться, определения на картинке:



- 64) **Гетерогенные** – системы объединяющие кластеры разных мощностей. Преимущества кластеров: 1) относительная дешевизна; 2) способность к расширению, увеличению мощностей.
- 65) **Терминальный комплекс** – это многомашинная ассоциация предназначенная для организации массового доступа удаленных и локальных пользователей к ресурсам некоторой вычислительной системы.
- 66) **Основная вычислительная система** – система, массовый доступ к ресурсам которой обеспечивается терминальным комплексом.
- 67) **Локальные мультимплексы** – аппаратные комплексы, предназначенные для осуществление связи и взаимодействия вычислительной системы с несколькими устройствами через один канал ввода/вывода.
- 68) **Локальные терминалы** – оконечные устройства, используемые для взаимодействия пользователей с вычислительной системой.
- 69) **Модемы** – устройства, предназначенные для организации взаимодействия вычислительной системы с удаленными терминалами с использованием телефонной сети. В функцию модема входит преобразование информации из дискретного, цифрового представления.
- 70) **Удаленные терминалы** – терминалы, имеющие доступ к вычислительной системе с использованием телефонных линий связи и модемов.
- 71) **Удаленные мультимплексы** – мультимплексы, подключенные к вычислительной системе с использованием телефонных линий связи и модемов.
- 72) **Виды каналов:** коммутируемые (каждый раз новый), выделенные(зарезервированный).
- 73) **Симплексные каналы** - каналы, по которым передача информации ведется в одном направлении
- 74) **Дуплексные каналы** - каналы, которые обеспечивают одновременную передачу информации в двух направлениях (например, телефонный разговор, мы одновременно можем и говорить и слушать).
- 75) **Полудуплексные каналы** - каналы, которые обеспечивают передачу информации в двух направлениях, но в каждый момент времени только в одну сторону (подобно рации).
- 76) **Компьютерная сеть** – объединение компьютеров (или вычислительных систем), взаимодействующих через коммуникационную среду.
- 77) **Коммуникационная среда** – каналы и средства передачи данных.
- 78) **Абонентские машины (хосты)** - обеспечивают обмен содержательной информацией работы с

пользователями.

- 79) **OSI** – системы открытых интерфейсов.
- 80) **Уровни взаимодействия компьютеров**: физический уровень, канальный уровень, сетевой уровень, транспортный уровень (уровень логического канала), сеансовый уровень, представительский уровень, прикладной уровень.
- 81) **Протокол** – формальное описание сообщений и правил, по которым сетевые устройства (вычислительные системы) осуществляют обмен информацией. Или правила взаимодействия одноименных уровней.
- 82) **Интерфейс** – правила взаимодействия вышестоящего уровня с нижестоящим.
- 83) **Служба или сервис** – набор операций, предоставляемых нижестоящим уровнем вышестоящему.
- 84) **Стек протоколов** – перечень разноуровневых протоколов, реализованных в системе
- 85) **Сообщение** — логически целостная порция данных, имеющая произвольный размер.
- 86) **Сеть коммутации каналов** — обеспечивает выделение коммутации связи на весь сеанс связи.
- 87) **Сеть коммутации пакетов** — обеспечивает выделение коммутации связи на передачу одного пакета.
- 88) **IP адрес** - последовательностью четырех байтов. В адресе кодируется уникальный номер сети, а также номер компьютера (сетевое устройство в сети).
- 89) **Пакет** – это блок данных, который передаётся вместе с информацией, необходимой для его корректной доставки. Каждый пакет перемещается по сети независимо от остальных.
- 90) **Дейтаграмма** – это пакет протокола IP. Контрольная информация занимает первые пять или шесть 32-битных слов дейтаграммы. Это её заголовок (header). По умолчанию, его длина равна пяти словам, шестое является дополнительным. Для указания точной длины заголовка в нём есть специальное поле – длина заголовка (IHL, Internal Header Length).
- 91) **Шлюз** – устройство, передающее пакеты между различными сетями.
- 92) **Маршрутизация** – процесс выбора шлюза или маршрутизатора
- 93) **Протокол контроля передачи (TCP, Transmission Control Protocol)** - обеспечивает надежную доставку данных с обнаружением и исправлением ошибок и с установлением логического соединения.
- 94) **Протокол пользовательских дейтаграмм (UDP, User Datagram Protocol)** - отправляет пакеты с данными, «не заботясь» об их доставке.
- 95) **Протоколы, опирающиеся на TCP** (TELNET (Network Terminal Protocol), FTP (File Transfer Protocol), SMTP (Simple Mail Transfer Protocol),
- 96) **Протоколы, опирающиеся на UDP** , DNS (Domain Name Service), RIP (Routing Information Protocol), NFS (Network File System)
- 97) **Сетевая ОС** – физическую сеть в которой подключенные компьютеры взаимодействуют с помощью протоколов, сетевая ОС предоставляет пользователям распределенные прикладные приложения
- 98) **Распределенная ОС** — ОС, функционирующая на многопроцессорном или многомашинном комплексе, в котором на каждом из узлов функционирует свое ядро, а также система, обеспечивающая распределение возможностей (ресурсов) ОС.
- 99) **Процесс** – это совокупность машинных команд и данных, обрабатываемых в рамках ВС и обладающая правами на владение некоторым набором ресурсов. (Вообще определений много)
- 100) Существуют ресурсы, которые монопольно принадлежат данному процессу, и **разделяемые ресурсы**, которые принадлежат одновременно 2 и более процессам.
- 101) **Выделения ресурсов процессу**: предварительная декларация использования тех или иных ресурсов, динамическое пополнение списка принадлежащих процессу ресурсов по ходу.

- 102) **Любая ОС должна удовлетворять набору свойств:** надежность, обеспечение защиты, эффективность, предсказуемость.
- 103) **Резидентная** – постоянно находящаяся в памяти
- 104) **Ядро (kernel)** – резидентная часть ОС, работающая в режиме супервизора. («обычно» работает в режиме физической адресации).
- 105) **Системный вызов** - средство ОС, обеспечивающее возможность процессов обращаться к ОС за теми или иными функциями.
- 106) **Монолитное ядро** – ядро, которое включает в себя все возможности операционной системы, запускаются как единый процесс.
- 107) Существует стационарное **микроядро**, которое обеспечивает минимальные функции ОС (на нем все держится)
- 108) **Логические функции ОС:** управление процессами, управление ОП, планирование, управление устройствами и ФС
- 109) **Буфер ввода процессов** - область на внешней памяти, где аккумулируются все процессы, которые еще не начали выполняться.
- 110) **Буфер обрабатываемых процессов** - область памяти, где хранятся данные процессов, которые начали обрабатываться в мультипрограммном режиме.
- 111) **Пакет программ** – совокупность программ, для выполнения каждого из которых требуется некоторое время работы процессора.
- 112) **Квант времени ЦП** – некоторый фиксированный ОС промежуток времени работы ЦП
- 113) **Системы реального времени** - являются специализированными системами в которых все функции планирования ориентированы на обработку некоторых событий за время, не превосходящее некоторого предельного значение
- 114) **Системы с разделением времени** – системы, в которых на выполнение каждого процесса отводится определенный промежуток (квант) процессорного времени.
- 115) **Считывание информации, алгоритмы: простейшая модель** – случайная выборка из очереди, FIFO, LIFO, SSTF – жадный, Приоритетный алгоритм (RPI) – это алгоритм, когда последовательность обменов (очередь) имеет характеристику приоритетов, SCAN, C-SCAN, N-step-SCAN.
- 116) **RAID система** представляет собой набор независимых дисков, которые рассматриваются ОС как единое дисковое устройство, где данные представляются в виде последовательности записей, которые называются полосы.
- 117) **Специальные файлы устройств** - в системе Unix единый интерфейс организации взаимодействия с внешними устройствами
- 118) **Файлы байториентированных устройств** (драйверы обеспечивают возможность побайтного обмена данными и, обычно, не используют централизованной внутрисистемной кэш-буферизации);
- 119) **Файлы блокориентированных устройств** (обмен с данными устройствами осуществляется фиксированными блоками данных, обмен осуществляется с использованием специального внутрисистемного буферного кэша).
- 120) **Индексный дескриптор** файла устройства содержит: тип файла устройства – байториентированный или блокориентированный, «старший номер» (major number) устройства - номер драйвера в соответствующей таблице драйверов устройств, «младший номер» (minor number) устройства – служебная информация, передающаяся драйверу устройства.
- 121) **bdevsw** – таблица драйверов блокориентированных устройств.
- 122) **cdevsw** - таблица байториентированных устройств.

- 123)**Существует два, традиционных способа включения драйверов новых устройств в систему:** путем «жесткого», статического встраивания драйвера в код ядра, требующего перекомпиляции исходных текстов ядра или пересборку объектных модулей ядра и за счет динамического включения драйвера в систему.
- 124)**Таблица индексных дескрипторов открытых файлов:** для каждого открытого в рамках системы файла формируется запись в таблице ТИДФ, содержащая: копии индексного дескриптора (ИД) открытого файла, кратность - счетчик открытых в системе файлов, связанных с данным ИД.
- 125)**Таблица файлов:** таблица файлов содержит сведения о всех файловых дескрипторах открытых в системе файлов.
- 126)**Таблица открытых файлов:** С каждым процессом связана таблица открытых файлов (ТОФ). Номер записи в данной таблице есть номер ФД, который может использоваться в процессе. Каждая строка этой таблицы имеет ссылку на соответствующую строку ТФ.
- 127)**Организация буферизации при обмене.** В RAM организуется пул буферов, где каждый буфер имеет размер в один блок. Каждый из этих блоков может быть ассоциирован с драйвером одного из физических блок
- 128)**SYNC** - по этой пользовательской команде осуществляется сброс данных на диск.
- 129)**Основные задачи при организации работы с ОП:** контроль состояния каждой единицы памяти (свободна/распределена), стратегия распределения памяти, выделение памяти, стратегия освобождения памяти.
- 130)**Стратегии и методы управления:** одиночное непрерывное распределение, распределение разделами, распределение перемещаемыми разделами, страничное распределение, сегментное распределение, сегменто-страничное распределение.
- 131)**TLB (Translation Lookaside Buffer)** – Буфер быстрого преобразования адресов.
- 132)**ХЭШ – функция** берет номер виртуальной страницы и по этому номеру виртуальной страницы имеется некоторая функция, которая определяет номер записи хэш-таблицы.
- 133)**Алгоритм NRU (Not Recently Used – не использовавшийся в последнее время)**
Используются биты статуса страницы. R – обращение, M – модификация. Устанавливаются аппаратно при обращении или модификации.
- 134)**Алгоритм FIFO** – first in, first out.
- 135)**Алгоритм LIFO** – last in, first out.
- 136)**Алгоритм LRU (Least Recently Used – «менее недавно»** - наиболее давно используемая страница)
- 137)**Алгоритм NFU (Not Frequently Used – редко использовавшаяся страница)**
- 138)**Файловая система (ФС)** - часть операционной системы, представляющая собой совокупность организованных наборов данных, хранящихся на внешних запоминающих устройствах, и программных средств, гарантирующих именованный доступ к этим данным и их защиту
- 139)Данные называются файлами, их имена - именами файлов.
- 140)**Правило работы с файлами:** открытие, работа с файлом, закрытие.
- 141)**Файловый дескриптор** – системная структура данных, содержащая информацию о актуальном состоянии «открытого» файла.
- 142)**Каталог** – компонент файловой системы, содержащий информацию о содержащихся в файловой системе файлах. (Каталог — это файл)
- 143)**Суперблок** – блок ФС, в котором находится информация о настройках ФС и актуальном состоянии ФС (информация о свободных блоках, данных, которые содержат каталоги.)
- 144)**Блок** – порция данных, фиксированного размера, в рамках которого идет обмен данными с устройством.

- 145) **Таблица размещения файловой системы** – таблица, в которой количество строк соответствует количеству блоков, в кот. i-ая строка соотв. i-ому блоку файловой системы.
- 146) **Индексные узлы или индексные дескрипторы** — организованное файловой системой компактное хранение информации о размещении блоков файлов в специальной структуре.
- 147) **“Жесткая” связь**: Есть содержимое файла, есть атрибут файла, и одним из полей атрибутов является количество имен у этого файла, и есть произвольное количество имен, которые как-то распределены по каталогам ФС.
- 148) **“Символическая” связь** есть файл с именем Name₂, этому имени соответствуют атрибуты и соответствует содержимое, и есть специальный файл Name₁, который ссылается на имя Name₂.
- 149) **Физическая архивация** («один в один» или интеллектуальная физическая архивация (копируются только использованные блоки файловой системы)
- 150) **Логическая архивация** – копирование файлов (а не блоков), модифицированных после заданной даты.
- 151) **Файл Unix** – это специальным образом именованный набор данных, размещенный в файловой системе.
- 152) **Обычный файл (regular file)** – традиционный тип файла, содержащий данные пользователя. Интерпретация содержимого файла производится программой, обрабатывающей файл.
- 153) **Каталог (directory)** – специальный файл, обеспечивающий иерархическую организацию файловой системы. С каталогом ассоциируются все файлы, которые принадлежат данному каталогу.
- 154) **Специальный файл устройств (special device file)** – система позволяет ассоциировать внешние устройства с драйверами и предоставляет доступ к внешним устройствам, согласно общим интерфейсам работы с файлами.
- 155) **Именованный канал (named pipe)** – специальная разновидность файлов, позволяющая организовывать передачу данных между взаимодействующими процессами;
- 156) **Ссылка (link)** – позволяет создавать дополнительные ссылки к содержимому файла из различных точек файловой системы; Они могут нарушать древовидность организации ФС.
- 157) **Сокет (socket)** – средство взаимодействия процессов в пределах сети ЭВМ.
- 158) **Корневой каталог /** является основой любой файловой системы ОС UNIX. Все остальные файлы и каталоги располагаются в рамках структуры, порожденной корневым каталогом, независимо от их физического положения на диске.
- 159) **/unix** - файл загрузки ядра ОС.
- 160) **/bin** - файлы, реализующие общедоступные команды системы.
- 161) **/etc** - в этом каталоге находятся файлы, определяющие настройки системы
- 162) **/tmp** - каталог для хранения временных системных файлов. При перезагрузке системы не гарантируется сохранение его содержимого.
- 163) **/mnt** - каталог, к которому осуществляется монтирование дополнительных физических файловых систем для получения единого дерева логической файловой системы. Заметим, что это лишь соглашение, в общем случае можно примонтировать к любому каталогу.
- 164) **/dev** - каталог содержит специальные файлы устройств, с которыми ассоциированы драйверы устройств.
- 165) **/lib** - здесь находятся библиотечные файлы языка Си и других языков программирования.
- 166) **/usr** - размещается вся информация, связанная с обеспечением работы пользователей. Здесь также имеется подкаталог, содержащий часть библиотечных файлов (/usr/lib), подкаталог /usr/users (или /usr/home), который становится текущим при входе пользователя в систему, подкаталог, где находятся дополнительные команды (/usr/bin), подкаталог, содержащий файлы заголовков (/usr/include), в котором,

в свою очередь, подкаталог, содержащий include-файлы, характеризующие работу системы (например, signal.h - интерпретация сигналов).

- 167) **Индексный дескриптор в ОС UNIX** – это специальная структура данных файловой системы, которая ставится во взаимно однозначное соответствие с каждым файлом.
- 168) **Процесс** - совокупность машинных команд и данных, которая исполняется в рамках вычислительной системы и обладает правами на владение некоторым набором ресурсов.
- 169) **Разделяемые ресурсы** - ресурсы разделяемые между процессами
- 170) **Буфер ввода процесса (БВП)** — пространство, в котором размещаются и хранятся сформированные процессы от момента их образования, до момента начала выполнения.
- 171) **Буфер обрабатываемых процессов (БОП)** — буфер, где размещаются процессы, работающие в мультипрограммном режиме.
- 172) **«Полновесные процессы»** - это процессы, выполняющиеся внутри защищенных участков памяти операционной системы, то есть имеющие собственные виртуальные адресные пространства для статических и динамических данных
- 173) **Легковесные процессы**, называемые еще как нити или сопрограммы, не имеют собственных защищенных областей памяти.
- 174) **Контекст процесса** - совокупность данных, характеризующих актуальное состояние процесса.
- 175) **Процесс в ОС Unix** – объект, зарегистрированный в таблице процессов Unix или объект, порожденный системным вызовом fork()
- 176) **Идентификатором процесса (PID)** - уникальным имя процесса. PID – целое число от 0 до некоторого предельного значения, определяющего максимальное число процессов (ресурс данной ОС), существующих в системе одновременно.
- 177) **Сегмент кода** содержит машинные команды и неизменяемые константы соответствующей процессу программы.
- 178) **Сегмент данных** – содержит данные, динамически изменяемые в ходе выполнения кода процесса.
- 179) **Системный вызов** – специальная функция, позволяющая процессу обращаться к ядру ОС за выполнением тех или иных действий.
- 180) **Семейство системных вызовов exec()** - заменяет тело вызывающего процесса, после чего данный процесс начинает выполнять другую программу. Управление передается на точку ее входа. Возврат к первоначальной программе происходит только в случае ошибки при обращении к exec(), т.е. если фактической замены тела процесса не произошло.
- 181) **Квант времени** – непрерывный период процессорного времени.
- 182) **Приоритет процесса** – числовое значение, показывающее степень привилегированности процесса при использовании ресурсов ВС (в частности, времени ЦП).
- 183) **Невытесняющая стратегия** - если величина кванта не ограничена
- 184) **Вытесняющая стратегия** - величина кванта ограничена.
- 185) **Простой круговорот (RR – round robin)** не использует никакой статистической или динамической информации о приоритетах.
- 186) При **круговороте со смещением** каждому процессу соответствует своя длина кванта, пропорциональная его приоритету.
- 187) **«Эгоистический» круговорот**. Если параметры A и B : $0 < B < A$. Процесс, войдя в систему ждет пока его приоритет не достигнет приоритета работающих процессов, а далее выполняется в круговороте. Приоритет выполняемых процессов увеличивается с коэффициентом $B < A$, следовательно, ожидающие процессы их догонят. При $B=0$ «эгоистический» круговорот практически сводится к простому
- 188) **Область свопинга** - специально выделенное системой пространство внешней памяти

- 189) **Параллельные процессы** - Процессы, выполнение которых хотя бы частично перекрывается по времени
- 190) **Независимые процессы** – процессы, использующие независимое множество ресурсов и на результат работы такого процесса не влияет работа независимого от него процесса.
- 191) **Взаимодействующие процессы** совместно используют ресурсы, и выполнение одного может оказывать влияние на результат другого.
- 192) **Критические ресурсы** — разделяемые ресурсы, которые должны быть доступны в текущий момент времени только одному процессу.
- 193) **Критическая секция, или критический интервал** - часть программы (фактически набор операций), в которой осуществляется работа с критическим ресурсом.
- 194) **Взаимное исключение** – т.е. такой способ работы с разделяемым ресурсом, при котором постулируется, что в тот момент, когда один из процессов работает с разделяемым ресурсом, все остальные процессы не могут иметь к нему доступ.
- 195) **Тупики (deadlocks)** - ситуации в которой конкурирующие за критический ресурс процессы безвозвратно блокируются.
- 196) **Блокирование** — ситуация, когда доступ одного из процессов к разделяемому ресурсу не обеспечивается из-за активности других, более приоритетных процессов.
- 197) **Семафоры** – это низкоуровневые средства синхронизации, для корректной практической реализации которых необходимо наличие специальных, атомарных семафорных машинных команд.
- 198) **Семафоры Дейкстры** — формальная модель, предложенная голландцем Дейкстрой, которая основывается на следующем предположении: имеется тип данных, именуемой семафором (только 2 значения). Над семафором определены 2 операции: увеличить, уменьшить.
- 199) **Монитор Хоара** — совокупность процедур и структур данных, объединенных в программный модуль специального вида.
- 200) **Сигнал** – средство уведомления процесса о наступлении некоторого события в системе. Инициаторы посылки сигнала - другой процесс или ОС.
- 201) **Неименованный канал**- область на диске, к которой не возможен доступ по имени, а только с помощью двух дискрипторов с ней ассоциированных.
- 202) **Именованный канал** - расширение понятия конвейера в Unix.
- 203) **Система IPC** – альтернатива именованным каналам. Позволяет организовывать работу именованных каналов в произвольные моменты времени.
- 204) **Очередь сообщений** представляет собой некое хранилище типизированных сообщений, организованное по принципу FIFO.
- 205) **Механизм разделяемой памяти** позволяет нескольким процессам получить отображение некоторых страниц из своей виртуальной памяти на общую область физической памяти.
- 206) **Семафоры** представляют собой одну из форм IPC и используются для синхронизации доступа нескольких процессов к разделяемым ресурсам, т.е. фактически они разрешают или запрещают процессу использование разделяемого ресурса.
- 207) **Сокет** – программный интерфейс для обеспечения информационного обмена между процессами.

Удачи!