1)Сукупність програм разом з документацією на них, що використовуються для автоматизації робіт по переробці інформації на комп'ютері, називають: {

~програмним забезпеченням;

~операційною системою;

~компілятором;

~ліцензійною програмою

~САПР;

}

2)Сукупність програм для керування ресурсами комп'ютера та організації взаємозв'язку з користувачем, називають: {

~програмним забезпеченням;

~операційною системою;

~прикладною програмою;

~інструментальним засобом;

~графічно оболонкою ОС;

}

3)Операційна система використовується для: {

~керування задачами;

~керування даними;

~організації інтерфейсу;

~керування пристроями;

~всіх перелічених пунктів;

}

4)В склад операційної системи входять: {

~інтерфейс користувача, командний процесор, драйвери;

~графічний редактор, драйвери, модулі програмування;

~драйвери, модулі програмування, інтерфейс;

~BIOS, системи програмування, мережа;

~драйвери, модулі програмування, командний процесор;

}

5)Підготовка вінчестера до початку встановлення ОС виконується програмою: {

~Fdisk;

~DiskEditor;

~Format;

~Install;

~Setup.

}

6)Встановлення сучасної операційної системи здійснюється тільки: {

~в основний розділ диска;

~в будь-який розділ диска;

~в пам'ять комп'ютера;

~в стартовий сектор диска;

~в стартовий розділ диска;

}

7)ОС напряму використовує : {

~оперативну пам'ять;

~постійну пам'ять;

~хешовану пам'ять;

~зовнішню пам'ять;

~всі перелічені;

}

8) Командним рядком ОС Windows є: {

~C:\\Linux&gt;;

~A:/ dfffn/hjffhdj/&gt;;

~home@www\_rrr$;

~всі перелічені;

~ні один з перелічених;

}

9)Структура сучасної операційної системи включає: {

~локальну частину, транспортну частину, клієнтську частину;

~серверну частину, транспортну частину, клієнтську частину;

~локальну частину, транспортну частину, мережеву частину;

~оперативну частину, серверну частину, клієнтську частину;

~оперативну частину, дискову частину, мережеву частину;

}

10)Програма, яка обслуговує деякий периферійний пристрій, називається: {

~драйвером;

~модулем;

~зовнішньою;

~внутрішньою;

~системою;

}

11)Сигнал,який поступає в процесор про здійснення асинхронної події, називають: {

~перериванням;

~асинхронним;

~блокованим;

~незалежним;

~синхронним;

}

12) Невеликий програмний код, що забезпечує завантаження з вінчестера однієї з відмічених спеціальним чином ОС є: {

~драйвером;

~BIOSом;

~позасистемним завантажувачем;

~сектором;

~системною програмою;

}

13) Розмір адресного простору n-розрядного процесора складає : {

~2<sup>n-1</sup>;

~n<sup>2</sup>;

~2<sup>n</sup>;

~8<sup>n</sup>.

~2n;

}

14)Підсистема введення-виведення включає в себе: {

~драйвери;

~файловий менеджер;

~операційну систему;

~системні області диска;

~командний процесор;

}

15) Структура дискових носіїв інформації включає в себе: {

~сектори, треки;

~каталоги, сектори;

~циліндри, файли;

~файли, каталоги;

~кластери, процеси;

}

16)У файлових система ОСWindows запис даних на диск здійснюється: {

~кластерами;

~секторами;

~байтами;

~файлами;

~тегами;

}

17) У каталог записують: {

~файли;

~інформацію про файли та підкаталоги;

~підкаталоги;

~текст;

~файли та підкаталоги;

}

18) В ОС Unix уніфікація поняття файлу означає наступне: {

~будь-які пристрої вважаються файлами;

~зовнішні пристрої не є файлами;

~спеціальні файли є каталогами ;

~у файлів немає розширення;

~процес може бути файлом;

}

19) Чотири основні секції: керування процесами, керування пам'яттю , керування даними (організація файлової системи ) та керування пристроями, складають: {

~ядро ОС Unix;

~підсистему введення-виведення;

~системні виклики ОС Unix;

~рівні захисту ОС Unix;

~секції функціонування ОС Unix;

}

20) Обчислення в межах одного процесу виконуються: {

~паралельно;

~розподілено;

~послідовно;

~віртуально;

~детально;

}

21) Реалізація свопінгу входить у функції : {

~диспетчерського процесу;

~користувацького процесу;

~термінального процесу;

~виконуваного процесу;

~переривання по свопінгу;

}

22) Інтерпретатор команд shell виконується після: {

~запуску ОС;

~запуску графічної оболонки;

~введення коректного імені та паролю користувача;

~запуску процесів з номерами 0, 1, 2;

~введення команд користувача;

}

23) Для кожного активного процесу створюється: {

~таблиця відповідності процесів;

~контекст процесу;

~хеш таблиця посилань процесу;

~таблиця файлів;

~активне переривання процесу;

}

24) Апарат подій доступний: {

~системним процесам;

~процесам ОС;

~прикладним програмам;

~демонам;

~всім процесам;

}

25) Вкажіть вірний варіант системних викликів для роботи з файловою системою: {

~ореn, сlose, delete, сгеаtе;

~ореn, сlose, геаd, write, сгеаtе;

~delete, remov, сгеаtе;

~ореn, сlose, геаd, write, rewrite;

~ореn, сlose, delete, remov, delete, сгеаtе;

}

26) Вкажіть вірний варіант для операції P над семафором S: {

~<b>Якщо</b> (S&lt;0) <b>то</b> (S:\~S+1) <b>інакше</b> (чекати на S);

~<b>Якщо</b> (один або більше процесів чекають на S) <b>то</b> (дозволити одному із процесів виконувати роботу)<b> інакше</b> (S:\=S+1);

~<b>Якщо</b> (S&gt;0) <b>то</b> (S:\=S-1) <b>інакше</b> (чекати на S);

~<b>Якщо</b> (один або більше процесів чекають на S) <b>то</b> (дозволити одному із процесів виконувати роботу)<b> інакше</b> (S:\=S-1).

~<b>Якщо</b> (один або більше процесів чекають на S) <b>то</b> (чекати на S);

}

27) Процес, який знаходиться в критичній області: {

~може бути заблокований;

~не може бути заблокований;

~не може виконуватися;

~може бути призупинений;

~не може здійснювати обмін даними;

}

28) Доступ до томів файлової системи здійснюється шляхом: {

~монтування до активної файлової системи;

~створення тому;

~переходу до вказаного файлу;

~відкриття файлової системи;

~підключення резервних віртуальних кластерів;

}

29) <i>Послідовністю команд</i> <b>С-shell</b> називається командний рядок, що складається з команд, розділених символом: {

~;

~:

~\\

~|

~\*

}

30) Семафори використовуються для: {

~синхронізації двох процесів;

~обміну даними між двома процесами;

~призупинки процесу;

~обміну даними між двома процесом та пам'яттю;

~відкриття потоку команд процесору;

}

31)Основною характеристикою ОС реального часу є: {

~робота в он-лайн режимі;

~наявність таймера;

~обчислення на етапі компіляції;

~діалоговий режим роботи;

~виконання команди за фіксований проміжок часу.

}

32) У сучасних операційних система кожен процес виконується : {

~в реальному режимі роботи процесора

~фіксований проміжок часу

~в системній фазі

~у власному адресному просторі

~в кільці захисту 0

}

33)Системний завантажувач завантажує: {

~Операційну систему

~Будь-яку програму

~Системну програму

~Тестову систему

~Програмне забезпечення

}

34)Позасистемний завантажувач завантажує: {

~Системний завантажувач

~Операційну систему

~Будь-яку програму

~Системну програму

~Тестову систему

}

35)BIOS це {

~Базова система введення-виведення

~Апаратне переривання

~Програмне забезпечення

~Операційна система

~Стартовий сектор

}

36)Системний диск це: {

~Диск, з якого можливе завантаження ОС

~Диск, на якому знаходиться ОС

~Диск, на якому знаходиться системне програмне забезпечення

~Флешка з ОС

~CD ROM

}

37)BIOS не містить: {

~драйвери нестандартних ПП

~тестові програми

~драйвери стандартних ПП

~програму початкового завантаження

~програму Setup

}

38)Драйвер це: {

~програма, яка обслуговує деякий периферійний пристрій

~тестова програма

~прикладна програма

~програма початкового завантаження

~керуючий пристрій

}

39) Переривання це {

~сигнал, який поступає в процесор про здійснення асинхронної події

~Апаратне переривання

~Програмне забезпечення

~Системний виклик

~Оператор мови програмування

}

40):Які події не викликають переривання: {

~Обчислення добутку двох чисел в програмах.

~події, які проходять в ПП. Процесор, який почав обмін з таким пристроєм, від імені одного процесу, може запустити інший процес. Після завершення обміну, ПП викличе переривання.

~визначені наперед події в процесах. Процес, який потребує послуг ОС, може викликати особливе переривання.

~непередбачувані події в процесах. При намаганні здійснення процесом невідомої або забороненої дії, проходить виклик переривання.

~дії оператора. Оператор, який бажає почати взаємодію з ОС, може викликати переривання.

}

41)Що називають процесом? {

=деяка послідовність дій, з яких складається задача

~Будь-яку програму

~Системну програму

~Тестову систему

~Програмне забезпечення

}

42)Підсистема введення-виведення {

=організує обмін інформацією з ПП

~виконує апаратне переривання

~систематизує програмне забезпечення

~здійснює системні виклики

~вводить та виводить дані в програму

}

43)Підсистема введення-виведення може не включати в себе: {

~Прикладну програму

~Драйвери накопичувачів на жорстких та гнучких магнітних дисках;

~Драйвери дисплея та клавіатури;

~драйвер принтера;

~драйвери адаптерів інтерфейсів (драйвери послідовних та паралельних портів);

}

44)Адресний простір це: {

~сукупність тих комірок пам'яті, які в стані адресувати процесор

~Системний завантажувач

~Адречи комірок пям'яті

~Віртуальна адресація

~Оперативна пям'ять

}

45)До функцій керування пам'яттю не належить: {

~Завантаження програми на виконання

~визначення найбільшого блока, який можна виділити програмі;

~виділення блока пам'яті заданого розміру;

~вивільнення блока пам'яті;

~зміни розміру вже виділеного блока пам'яті як в сторону збільшення, так в сторону зменшення.

}

46)Стандартний розмір сектора: {

~512 байт

~1012 байт

~1000 байт

~512 Кбайт

~512 Мбайт

}

47)Системна область диска містить: {

~Таблицю розміщення файлів

~1024 байти

~Вміст каталога

~Інформацію про файли

~Атрибути файлів

}

48)Яке ім'я файлу не є стандартним: {

~COMP

~AUX

~CON

~LPT

~NUL

}

49)Який набір символів використовують для задання шаблонів імен файлів: {

~\*, ?

~?, +, -, \*, /

~+, -, \*, /

~?, +, \_, \*, /

~?, +, -, \*, \_

}

50)<b>Шлях</b> це: {

~назва диску та послідовність імен каталогів

~відносна адреса файлу

~командний інтерпретатор

~послідовність імен файлів у каталозі

~назви підкаталогів у поточному каталозі

}

51)<b>Елемент каталогу ..</b> це: {

~Позначення каталогу вищого рівня

~відносна адреса файлу

~позначення імені файлу

~позначення самого каталогу

~назва підкаталогу

}

52)Файлом називають: {

~Область даних на диску

~Базову систему введення-виведення

~Програмне забезпечення

~Елемент каталогу

~Кластери на диску

}

53)

? в шаблонах імен файлів позначає: {

~Тільки один будь-який символ

~Будь-яку кількість символів

~Елемент каталогу

~Кластер на диску

}

54)\* в шаблонах імен файлів позначає: {

~Будь-яку кількість будь-яких символів символів

~Тільки один будь-який символ

~Елемент даних

~Елемент каталогу

~Кластер на диску

}

55)Розширення імені файлу, як правила, вказує на: {

~Тип інформації у файлі

~Область даних на диску

~Програмне забезпечення

~Порядок сортування

~Сектор на диску

}

56)Каталогом називають: {

~Спеціальний тип файлу, в якому реєструють інші файли та каталоги

~Область даних на диску

~Системне програмне забезпечення

~Місце, де зберігаються файли, та яке має і'мя й інші атрибути

~Кластери на диску

}

57)До основних концепцій побудови ОС Unix не відноситься: {

~Конгруєнтність

~Модульність

~Мобільність

~Керування процесами

~Комунікація процесів

}

58)До основних концепцій побудови ОС Unix не відноситься: {

~Утиліти

~Модульність

~Уніфікація поняття файлу

~Командна мова

~Комунікація процесів

}

59)До основних концепцій побудови ОС Unix не відноситься: {

~Ядро системи

~Керування процесами

~Уніфікація поняття файлу

~Організація файлової системи

~Комунікація процесів

}

60)До основних концепцій побудови ОС Unix не відноситься: {

~Задача "голодного філософа"

~Керування процесами

~Інструментальний характер

~Організація файлової системи

~Мобільність

}

61)До основних концепцій побудови ОС Unix не відноситься: {

~Ядро системи

~Керування процесами

~Уніфікація поняття файлу

~Організація файлової системи

~Комунікація процесів

}

62)Яку операцію не можна виконувати над файлом: {

~переадресація

~читання

~запис

~знищення

~закриття

}

63)Файлову систему зображують у вигляді: {

~дерева

~незв'язаного списку

~куща списків

~файлової системи каталогів

~Комунікації пристроїв

}

64)До основних компонент ОС Unix не відносять: {

~Текстовий редактор Word

~Ядро

~утиліти

~Програми системного адміністратора

~Засоби розробки програм

}

65)До основних компонент ОС Unix не відносять: {

~Каталоги із файлами

~LEX

~утиліти

~Програми обробки текстів

~Засоби розробки програм

}

66)До основних компонент ОС Unix не відносять: {

~Процеси та сигнали

~YACC

~Файли- інструкції користувача

~Програми обробки текстів

~Програми організації інтерфейсу

}

67)Ядро ОС Unix не містить секції: {

~Кластерів

~Керування процесами

~Керування даними

~Керування пристроями

~Керування пам'яттю

}

68)Процесом називають: {

~Програму на стадії виконання

~Будь-яку послідовність операторів

~Виконуючу програму на диску

~Диспетчер задач

~Послідовність команд процесора

}

69)Диспетчерський процес в ОС Unix запускається: {

~Першим

~Останнім

~Другим

~Третім

~Одночасно з іншими процесами

}

70)Диспетчерський процес при запуску ОС Unix породжує наступний процес: {

~init

~format

~ttyys

~/etc

~admin

}

71)Для кожного процесу створюється таблиця: {

~Контексту процесу

~Використаних даних з оперативної пям'яті

~розміщення файлів

~Створених терміналів

~Віртуальних адрес

}

72)тКонтекст процесу не містить: {

~Таблицю розміщення файлів

~Номер користувача

~Посилання на системну таблицю файлів

~Номер групи користувачів

~Список реакцій на сигнали

}

73)Контекст процесу не містить: {

~Таблицю розміщення файлів

~Номер користувача

~Посилання на системну таблицю файлів

~Номер групи користувачів

~Список реакцій на сигнали

}

74)Основні стани процесів: {

~Готовий, блокований, призупинений

~Готовий, виконаний, призупинений

~Виконаний, блокований, призупинений

~Готовий, блокований, призупинений, виконаний

~Системний, блокований, виконаний

}

75)Свопінг це: {

~Вивантаження з ОП та завантаження в ОП процесів

~Обмін даними між програмами

~Запис на диск свінгерів

~Виконання реентерабельних програм

~Системний виклик процесів

}

76)Процес вважається системним: {

~З моменту системного виклику і до його завершення

~З моменту запуску операційною системою

~З моменту запуску диспетчером задач

~З моменту виконання реентерабельних програм

~З моменту запису у файл свопінгу

}

77)Механізм подій доступний : {

~Системним процесам

~Процесам користувача

~Прикладним програмам

~Реентерабельним програмам

~Файловій системі

}

78)Сегменти є наступних типів : {

~Процедурним, динамічним, даних

~Процедурним, динамічним, даних, адрес, фрагментів

~Процедурним, динамічним, процесорним

~Процесним, динамічним, даних

~Процедурним, процесорним

}

79)Процеси виконуються у : {

~Власному віртуальному адресному просторі

~Власному реальному адресному просторі

~Власному віртуальному файловому просторі адрес

~Сумісному віртуальному адресному просторі

~Власному захищеному парольному режимі

}

80)Реенерабельна програма : {

~Програма з повторним входом

~Програма з повторним виходом

~Програма з реестром

~Програма з перезаписуваним простором адрес

~Програма з рентабельним терміном використання

}

81)POSIX це : {

~Міжнародний стандарт на машиннонезалежний інтерфейс комп'ютерного середовища

~Програма з повторним виходом, виходом, 10-ти кратним виконанням

~Програма з реестром

~Стандарт на операційні системи

~Протокол міжнародного стандарту роботи в мережі

}

82)Механізм подій: {

~Не дає можливості передавати дані

~Дає можливість передавати дані

~Програмувати події передавання даних

~Програма з перезаписуваним простором адрес

~Призначений для реагування користувача на події

}

83)Сигнал SIGINT означає: {

~Переривання

~Невірну машинну інструкцію

~Завершення процесу

~Сигнал про інтегральне обчислення

~Реагування користувача на подію

}

84)Сигнал SIGILL означає: {

~Невірну машинну інструкцію

~Переривання-пастку

~Завершення процесу

~Сигнал про логічну операцію

~Іллегальні процеси та інструкції

}

85)Сигнал SIGKILL означає: {

~Знищення процесу

~Переривання-пастку

~Кількість машинних інструкцій

~Аварійний вихід

~Аваційний вхід

}

86)Сигнал SIGALRM означає: {

~Переривання за таймером

~Переривання

~Сигнал тривоги

~Сигнал про термінове обчислення

~Невірну машинну інструкцію

}

87)Процедурний сегмент містить: {

~Машинні інструкції та константи

~Дані

~Процедури та функції

~Адресний простір процедур

~Статичні дані процедур

}

89)Динамічний сегмент містить: {

~Дані, що не ініціалізуються при компіляції

~Машинні інструкції та константи

~Процедури та функції

~Адресний простір процедур

~Статичні дані процедур

}

90)Сегмент даних містить: {  
~Дані, що ініціалізуються при компіляції

~Машинні інструкції та константи

~Процедури та функції

~Адресний простір процедур

~Статичні дані файлових систем

}

91)Виконуваний файл містить: {

~Процедурний сегмент, сегмент даних

~Процедурний сегмент, динамічний сегмент, сегмент даних

~Процедури та функції, оператори

~Адресний простір процедур, оператори

~Статичні дані, оператори

}

92)Корінь дерева каталогів в ОС Unix позначають: {

~/

~\\

~C:\\

~c/

~//

}

93)Суфікси файлі в ОС Unix позначають: {

~Зміст інформації у файлі

~Завершальна частина імені файлу

~Потрійну адресацію

~Прив'язку до кореня

~Основні командні скрипти

}

94)При записі шаблонів імен файлів в ОС Unix не використовують: {

~/

~\*

~?

~[

~]

}

95)Виконуючі файли в ОС Unix: {

~Як правило, суфіксів не мають

~мають суфікс ехе

~мають суфікс сом

~мають суфікс bin

~мають два суфікси

}

96)Абсолютним шляхом до файлу в ОС Unix називають: {

~Шлях, який починається з кореня

~Шлях, який починається з активного каталогу

~Шлях, який починається з каталогу /abs

~Послідовність імен каталогів доступу до утиліт

~Шлях, який починається з пасивного каталогу

}

97)Відносним шляхом до файлу в ОС Unix називають: {

~Шлях, який починається з активного каталогу

~Шлях, який починається з кореня

~Шлях, який починається з каталогу /abs

~Послідовність імен каталогів доступу до утиліт

~Шлях, який починається з пасивного каталогу

}

98)Спеціальні файли в ОС Unix позначають: {

~Пристрої введення-виведення

~Шлях до спеціального пристрою

~Файл із спеціальними системними функціями

~Послідовність імен каталогів кожного тому

~Процеси, процесори, стани процесів

}

99)Спеціальному файлу в ОС Unix поставлено у відповідність: {

~Номер пристрою та номер типу пристрою

~Номер спеціальної функції та номер підпрограми

~Файли із спеціальними системними функціями

~Ідентифікатори користувачів файлі

~Процеси, що використовують спеціальні файли

}

100)Спеціальні файли в ОС Unix: {

~Не містять символьної інформації

~Містять символьну інформацію

~Містять числову інформацію

~Містять інформацію про користувачів

~Містять інформацію про паролі користувачів

}

101)Дозвіл на виконання каталогу в ОС Unix означає: {

~Пошук з метою формування повного імені файлу

~Виконання всіх програм каталогу

~Виконання однієї програми, за вибором користувача

~Дозвіл на зміну атрибутів фалів у каталозі

~Запит паролю користувача цього каталогу

}

102)Атрибути файлів в ОС Unix: {

~Визначають права власника, групи власників, інших осіб

~Визначають права власника, групи власників, адміністратора

~Визначають права власника, roota, адміністратора

~Визначають права роботи в мережі з файлами

~Містять інформацію про користувачів файлів

}

103)Файл в ОС Unix може: {

~Мати два однакових імені

~Мати тільки одне імя

~Міститися на двох розділах

~Мати два однакових імені в одному каталозі

~Не мати імені

}

104)Файлові системи в ОС Unix: {

~Монтуються до визначеного каталогу

~Позначаються латинськими літерами

~Встановлюються в оперативну память

~Знаходяться в оперативній памяті

~Не монтуються до каталогів користувачів

}

105)Каталог /etc в ОС Unix містить: {

~Команди адміністратора системи

~Інші команди користувачів

~Всі командні файли

~Прикладні встановлені програми

~Відкомпільовані програми користувачів

}

106)Каталог /dev в ОС Unix містить: {

~Каталоги спеціальних файлів, файли роботи із пристроями

~Команди адміністратора системи

~Всі файли встановлених пристроїв та програм

~Прикладні встановлені програми

~Відкомпільовані програми користувачів

}

107)Суперблок містить інформацію про: {

~Фізичну структуру файлової системи

~Логічну структуру файлів

~Найважливіші дані про завантажувач

~Завантажувач операційних систем

~Зареєстрованих системних адміністраторів

}

108)Індексний дескриптор файлу в ОС Unix не містить інформації про: {

~Команди адміністратора системи

~Режим доступу та тип файлу

~Дату створення файлу

~Кількість посилань на файл

~Список вказівників на блоки даних

}

109)Імя файлу в ОС Unix міститься в: {

~Елементі каталогу

~Індексному дескрипторі файлу

~Суперблоці

~Посиланні на файл

~Кластері

}

110)Обсяг індексного дескриптора файлу в ОС Unix: {

~64 б

~12 Кб

~512 б

~4 Мб

~1024 б

}

111)Для адресації блоків файлу в ОС Unix використовують: {

~Список з 13 елементів

~FAT

~Список з 12 елементів

~512 бітну адресу

~Список вказівників на блоки даних

}

112)11 елемент списку адрес блоків файлу в ОС Unix є: {

~Адресою блоку із 128 адрес блоків даних

~Адресою блоку із 128 адрес непрямих блоків

~Списком з 12 елементів

~Адресою блоку із 128 адрес подвійних непрямих блоків

~Адресою блоку із 128 адрес потрійних непрямих блоків

}

113)13 елемент списку адрес блоків файлу в ОС Unix є: {

~Адресою блоку із 128 адрес потрійних непрямих блоків

~Адресою блоку із 128 адрес непрямих блоків

~Списком з 12 елементів

~Адресою блоку із 128 адрес подвійних непрямих блоків

~Адресою блоку із 128 адрес блоків даних

}

114)12 елемент списку адрес блоків файлу в ОС Unix є: {

~ Адресою блоку із 128 адрес подвійних непрямих блоків

~Адресою блоку із 128 адрес непрямих блоків

~Списком з 12 елементів

~Адресою блоку із 128 адрес блоків даних

~Адресою блоку із 128 адрес потрійних непрямих блоків

}

115)Командним рядком в ОС Unix називають: {

~Послідовність символів, що закінчується символом return

~Рядок тексту у скрипті

~Рядок тексту у bat-файлі

~Набраний текст у рядку введення

~Адресою машинної команди

}

116)Командний рядок в ОС Unix не містить: {

~Код машинної команди

~Просту команду

~Послідовність команд

~Групу команд

~Метасимволи

}

117)Послідовністю команд в ОС Unix називають: {

~Команди, розділені символом ;

~Команди, розділені символом |

~Команди, розділені символом ..

~Команди, розділені символом &amp;&amp;

~Команди, розділені метасимволами

}

118)Конвеєр в ОС Unix організовують за допомогою символу: {

~|

~;

~..

~&amp;&amp;

~:

}

119)Переадресація виведення в ОС здійснюється за допомогою символів: {

~&gt;, &gt;&gt;

~; , -&gt;

~..

~&amp;&amp;

~&lt;:&gt;, &lt;<--

}

120)Конвеєр в ОС Unix організовують за допомогою символу: {

~|

~;

~..

~&amp;&amp;

~:

}

121)Внутрішні змінні в ОС Unix : {

~Встановлюють режим роботи інтерпретатора

~Встановлюють числові значення параметрів операційної системи

~Визначаються користувачем

~Вводяться при запуску операційної системи

~Виводять на екран параметри роботи ОС

}

122)Зовнішні змінні в ОС Unix : {

~Аналогічні звичайним змінним мов програмування

~Встановлюють числові значення параметрів операційної системи

~Вводяться при запуску операційної системи

~Встановлюють режим роботи інтерпретатора

~Виводять на екран параметри роботи ОС

}

123)Підстановка значень змінних в ОС Unix відбувається: {

~В тому випадку, коли перед нею стоїть символ $

~В тому випадку, коли перед нею стоїть символ return

~В тому випадку, коли вона використовується в операторі

~В тому випадку, коли здійснюється виведення на екран

~При зверненні до неї

}

124)При описі імен файлів аргументів команд в ОС Unix можна використовувати: {

~Метасимволи

~Лише малі літери

~Звертання до елементів каталогу

~Назви утиліт

~Внутрішні змінні

}

125)Підстановка значень змінних в ОС Unix відбувається: {

~В тому випадку, коли перед нею стоїть символ $

~В тому випадку, коли перед нею стоїть символ return

~В тому випадку, коли вона використовується в операторі

~В тому випадку, коли здійснюється виведення на екран

~При зверненні до неї

}

126)

Команда для зміни каталогу в ОС Unix : {

~cd

~nam

~dir

~fg

~cat

}

127)

Команда для виведення тексту в ОС Unix : {

~echo

~nam

~dir

~fg

~text

}

128)

Команда для виходу з інтерпретатора в ОС Unix : {

~exit

~return

~tar

~fag

~break

}

129)

Команда для виведення таблиці процесів в ОС Unix : {

~jobs

~process

~write

~output

~proctabl

}

130)

Команда для завершення процесів в ОС Unix : {

~kill

~processout

~close

~output

~jobs

}

131)

Команда для виконання запиту входу в ОС Unix : {

~login

~users

~open

~ps

~jobs

}

132)

Команда для завершення роботи користувача в ОС Unix : {

~logout

~outusers

~close

~pass

~login

}

133)

Команда для підрахунку кількості рядків у файлі в ОС Unix : {

~wc

~outstroc

~summ

~calcstr

~who

}

134)

Команда для видачі інформації про активні процеси в ОС Unix : {

~ps

~outuproc

~write

~paswd

~loginproc

}

135)

Команда для отримання статусу особливого користувача (root) в ОС Unix : {

~su

~login

~cls

~passwd

~admin

}

136

Команда для синхронізації запису системних буферів в ОС Unix : {

~sync

~byffers

~tag

~far

~logttybyf

}

137

Команда для виведення повного імені поточного каталогу в ОС Unix : {

~pwd

~outupwd

~passwd

~prdir

~ls

}

138

Команда для списку файлів в ОС Unix : {

~ls

~outudir

~katalog

~folder

~print

}

139

Команда для створення каталогу в ОС Unix : {

~mkdir

~mkfolder

~opendir

~makefolder

~dirgen

}

140

Команда для знищення каталогу в ОС Unix : {

~rmdir

~rmfolder

~closedir

~makefolder

~mkdir

}

141

Команда для копіювання файлу в ОС Unix : {

~cp

~rmfile

~copyfile

~cat

~pwd

}

142

Команда для знищення файлу в ОС Unix : {

~rm

~rmfile

~rmdir

~kill

~pewd

}

143

Команда для створення архівного файлу в ОС Unix : {

~tar

~catarj

~setarx

~gzip

~cdzip

}

144

Команда для створення нового імені файлу в ОС Unix : {

~ln

~lsfile

~ls

~create

~tag

}

145

Команда для знищення файлу в ОС Unix : {

~rm

~rmfile

~rmdir

~kill

~pewd

}

146

Зміну стану процесів викликають : {

~події

~системні процедури

~команда tar

~прикладні програми

~трансферні процедури

}

147

Процес заблокований якщо : {

~Очікує появи події

~Записаний у файл свопінгу

~Блокується іншим процесом

~Знятий з виконання

~Отримав повідомлення про настання події

}

148

Процес виконується якщо : {

~Йому виділений процесор

~Записаний у файл свопінгу

~Запущений диспетчером задач

~Очікує появи події

~Отримав повідомлення про настання події

}

149

Список процесів, що готові до виконання : {

~Впорядковані за пріоритетами

~Не впорядковані за пріоритетами

~Не відсортовані

~Очікують появи події

~Отримають повідомлення про настання події

}

150

Список процесів, що заблоковані : {

~Не впорядковані за пріоритетами

~Впорядковані за пріоритетами

~Відсортовані

~Не очікують появи події

~Отримають повідомлення про настання події

}

151

Розблокування процесів здійснюється : {

~В тому порядку, в якому відбуваються очікувані ними події

~Самим заблокованим процесом

~Прикладною програмою

~Після завершення кванту часу

~Командами користувача

}

152

Переведення в стан готовності виконуваного процесу здійснюється : {

~Після завершення кванту часу

~Сигналом про подію в периферійному пристрої

~Прикладною програмою

~Після завершення події

~Командами користувача

}

153

Блок керування процесом не містить інформації про : {

~Очікувані події

~Поточний стан процесу

~Ідентифікатор процесу

~Виділені ресурси

~Область збереження регістрів

}

154

Над процесами не виконують наступну операцію : {

~Запис в Область збереження регістрів

~Створення

~Блокування

~Запуск

~Призупинка

}

155

Створення процесу не передбачає : {

~Запис в Область збереження регістрів

~Присвоєння імені процесу

~Визначення початкового пріоритету

~Формування PCB

~Виділення початкових ресурсів

}

156

Команда для створення каталогу в ОС Windows : {

~md

~mkfolder

~opendir

~makefolder

~dirgen

}

157

Команда для переміщення файлів в ОС Windows : {

~move

~mkfiles

~vol

~path

~dirgen

}

158

Команда для перегляду каталогу в ОС Windows : {

~dir

~ls

~opendir

~ln

~dirls

}

159

Команда для форматування диска в ОС Windows : {

~Format d:

~Format /p

~Format /?

~fdisk

~format FAT

}

160

Команда для перегляду вмісту файлу в ОС Windows : {

~type

~man

~cat

~more

~dirfile

}

161

Команда для знищення файлу в ОС Windows : {

~del

~remove

~kill

~more

~filedel

}

162

Команда для очищення консолі в ОС Windows : {

~Cls

~remove

~del

~more

~Conclear

}

163

Перехід на мітку в bat- файлі в ОС Windows виконується командою: {

~goto

~label

~label goto

~for

~exit

}

164

Вибір дій в bat- файлі в ОС Windows виконується командою: {

~Choice

~Case

~If

~for

~Ren

}

165

Перевірка умови в bat- файлі в ОС Windows виконується командою: {

~If

~Case

~Choice

~While

~Ren

}

166

Мітка в bat- файлі в ОС Windows починається із символа: {

~:

~Будь-якої букви

~\*

~Цифри

~@

}

167

Віртуальна машина це: {

~модель обчислювальної машини, створеної шляхом віртуалізації обчислювальних ресурсів: процесора, оперативної пам'яті, пристроїв зберігання та вводу і виводу інформації

~Математичний опис реального компютера

~Аналог віртуальної адреси

~Прикладне програмне забезпечення

~Програмний опис диспетчера задач

}

168

Що не є віртуальною машиною: {

~Android

~VirtualBox

~VMware

~IBM VM

~Java VM

}

169

При форматуванні носіїв інформації не виконується наступна дія: {

~Розбивка вінчестера на розділи:

~Пошук дефектних секторів

~Форматування системних областей

~Запис поверхні спеціальним кодом

~Розбиття поверхні на сектори

}

170

В каталозі немає наступної інформації про файл (FAT): {

~Номер розділу:

~Імя файлу

~Байт атрибутів

~Дата створення файлу

~Номер початкового кластера з даними

}

// Начало вопроса: 06. ВопрМножВыбор

Файлова система FAT32: {

~Підтримує "довгі" імена файлів:

~Не містить кореневого каталогу

~Розміщується тільки на 32-бітних носіях інформації

~Записує дату створення файлу кирилицею

~Записує файли в кореневий каталог

}

"Довгі" імена файлів не можуть містити символа(ів): {

~\*

~. !

~№1

~$

~.\_.

}

Файлова система FAT32 включена в ОС: {

~Windows

~Android

~MS DOS

~Unix

~Linux

}

// Начало вопроса: 06. ВопрМножВыбор

Файлова система NTFS включена в ОС: {

~Windows

~Android

~MS DOS

~Unix

~Linux

}

// Начало вопроса: 06. ВопрМножВыбор

Файлова система NTFS не містить: {

~Таблицю розділів:

~Метафайли

~Кластери

~Мета-зону

~Область даних

}

Найважливіший файл NTFS : {

~$MFT:

~Confyg.sys

~Kernel.exe

~swap

~GDI.exe

}

Призначення файлу $MFT: {

~Спільна таблиця файлів

~Підтримка Журналювання

~Кореневий каталог

~Карта вільного місця

~Завантажувальний сектор

}

Призначення файлу $Boot: {

~Завантажувальний сектор

~Права користувачів

~Кореневий каталог

~Карта вільного місця

~Спільна таблиця файлів

}

Призначення файлу $Upcase: {

~Таблиця відповідності великих і малих літер

~Підтримка Журналювання

~Верхня сторінка Кореневого каталогу

~Карта вільного місця

~Спільна таблиця файлів та Завантажувальний сектор

}

Призупинений процес активізується в наступному випадку: {

~Його активізує інший процес

~Знищиться РСВ

~Самоактивізується

~Завершиться операція введення-виведення

~Виконається команда active proc

}

Призупинка процесу потрібна в наступному випадку: {

~Пікового навантаження на систему

~Знищення процесу

~Виникнення переривання

~Перерозподілу оперативної памяті

~Запуску системного процесу

}

При виникненні переривання: {

~Керування передається ОС

~Знищується РСВ

~Файл записується на диск

~Виконується наступний користувацький процес

~Зупиняється процесор

}

В ієрархію ОС не включають: {

~Файли на носіях інформації

~Апаратуру

~Функції ядра

~Процеси ОС, Які підтримують процеси користувача

~Процеси користувача

~Асинхронними процесами називають процеси:

}

Які процеси не існують одночасно {

~Виконуючі процеси

~Призупинені процеси

~Блоковані процеси

~Спільні процеси, що створюються процесом-батьком

~Готові до виконання

}

Взаємовиключення це: {

~Виключення одночасного доступу до розділюваних змінних

~Виключення одночасного доступу до файлів

~Виключення одночасного доступу до принтера

~Виключення ділення на 0

~Паралельна робота процесорів

}

Розділювані змінні це: {

~Змінні, які одночасно можуть використовуватися декількома процесами

~Імя яких складається з двох частин

~Змінні, поділені на 2

~Виключені змінні із процесів

~Змінні, які ділять віртуальних адресний простір

}

Критична область процесу: {

~Область, в якій процес використовує розділювачу змінну

~Область, в якій процес використовує критичні операції

~Область, в якій процес виконує ділення на 0

~Область, в якій процес здійснює системний виклик

~Область, в якій процес паралельно працює з іншими процесами

}

Команда testandset(a, b): {

~Зчитує значення змінної b, копіює її в a, встановлює значення b в істину

~Зчитує значення змінної b, копіює її в a, встановлює значення b в хибність

~Зчитує значення змінної b, копіює її в a

~Обмінює значення змінних b та a

~Обнуляє значення змінних b та a

}

Семафор це: {

~Захищена змінна, значення якої змінюється операціями P та V

~Не захищена змінна, значення якої змінюється операціями P та V

~Тип даних процесора

~Оператор над захищеними змінними

~Змінна, яка може приймати три значення

}

Операція P над семафором S: {

~Якщо S&gt;0 то S\~S-1 інакше чекати на S

~Якщо один або більше процесів чекають на S то дозволити одному із процесів продовжити роботу, інакше S\=S+1

~Якщо S&gt;\=0 то S\=S+1 інакше чекати на S

~Якщо S&lt;0 то S\=S+1 інакше чекати на S

~Якщо процеси не чекають на S то призупинити процес, інакше S\=S-1

}

Операція V над семафором S: {

~Якщо один або більше процесів чекають на S то дозволити одному із процесів продовжити роботу, інакше S\=S+1

~Якщо S&gt;0 то S\=S-1 інакше чекати на S

~Якщо S&gt;\=0 то S\=S+1 інакше чекати на S

~Якщо S&lt;0 то S\=S+1 інакше чекати на S

~Якщо процеси не чекають на S то призупинити процес, інакше S\=S-1

}

Операції P та V: {

~неподільні

~взаємоповязані

~доповнюють один одного

~критичні

~гарантують виконання процесу

}

Операції P та V: {

~Реалізуються апаратно або в ядрі ОС

~Реалізуються в прикладних програмах

~Реалізуються в програмах користувача

~Не реалізовані в компютерах

~Використовуються тільки в алгоритмах

}

Зчитуючі семафори приймають значення: {

~Цілих невідємних чисел

~Нуль та один

~На множині дійсних чисел

~Символів латинського алфавіту

~Введених цілих чисел

}

Двійкові семафори приймають значення: {

~Нуль або один

~Цілих невідємних чисел

~Бінарних чисел

~Кодів символів

~Введених послідовностей нулів та одиниць

}

Семафори використовують для : {

~Синхронізації двох процесів

~Обміну даними

~Введення даних

~Виведення інформації

~Перевірки станів процесів

}

Зчитуючі семафори використовують для : {

~Організації виділення однорідного ресурсу із загального пулу

~Обміну даними

~Введення даних

~Виведення інформації

~Перевірки зчитаних символів

Перший параметр в bat- файлі в ОС Windows записується як: {

~ % 1

~ % % 1

~ % 1 %

~ Goto % 1

~ 1

}