

### **1. Назовите 4 причины необходимости изучения ОС**

Есть несколько причин, необходимость изучения операционных систем, наиболее важным из которых:

- для использования hardware для специального назначения, возможно, потребуется создать собственную операционную систему или внести изменения в существующую;
- Правильный выбор операционной системы и конкретного варианта может зависеть от жизнеспособности и эффективности расчетной системы;
- идеально подходит пользователю взаимодействовать с операционной системой, зная все её тонкости, потому что операционная система является посредником между компьютером и пользователем;
- Многие методы и понятия, используемые в области ОС, могут быть успешно реализованы в других областях.

### **2. Что называем мультипрограммированием?**

Мультипрограммирование — это способ организации вычислительного процесса, при котором на одном процессоре попеременно выполняются сразу несколько программ. Эти программы совместно используют не только процессор, но и другие ресурсы компьютера: оперативную и внешнюю память, устройства ввода-вывода, данные. Мультипрограммирование призвано повысить эффективность использования вычислительной системы. Наиболее характерными критериями эффективности вычислительных систем являются:

- пропускная способность — количество задач, выполняемых вычислительной системой в единицу времени;
- удобство работы пользователей, заключающееся, в частности, в том, что они имеют возможность интерактивно работать одновременно с несколькими приложениями на одной машине;
- реактивность системы — способность системы выдерживать заранее заданные (возможно, очень короткие) интервалы времени между запуском программы и получением результата.

### **3. Назовите 4 функции которые должна выполнить компонента управления процессором.**

ОС - это неотъемлемая часть ПО, управляющая техническими средствами компьютера (hardware)..

Операционная система - это программа, координирующая действия вычислительной машины; под её управлением осуществляется выполнение программ.

Основные функции операционной системы

- 1. Обмен данными между компьютером и различными периферийными устройствами (терминалами, принтерами, гибкими дисками, жесткими дисками и т.д.). Такой обмен данными называется "ввод/вывод данных".
- 2. Обеспечение системы организации и хранения файлов.
- 3. Загрузка программ в память и обеспечение их выполнения.
- 4. Организация диалога с пользователем.

### **4. В чем суть механизма отклонений?**

В отличие от прерывания, отклонение или вызов наблюдателя обусловлен с выполнением текущей команды. Отклонение (trap, англ..) Указывает на ненормальность в развитии инструкции, и выполнения команд. Происхождение может быть разнообразным:

Неверные данные, что приводит к невозможности корректного выполнения команды (деление на ноль, например), попытка выполнения операции, запрещенной механизмом защиты (нарушение защиты памяти, режим привилегированного команд программы и т.д.) неисполняемая инструкция (несанкционированное использование кода, использование дополнительного устройства, которое в данной конфигурации отсутствует и т.д.).

Отклонения могут быть классифицированы, как и прерывания, в соответствии с причинами которые их генерируют. Эффект какой либо причины может быть отменен (например, арифметические операции, связанные отклонениями, ошибки будут включены только в значение условия кода).

Важно! отклонение не позволяет применение концепции маскировки. Отклонение может быть подавлено.

### **5. Что называем строго иерархической реализацией ОС?**

В концепции многоуровневой (многослойной) иерархической машины структура ОС также представляется рядом слоев. При такой организации каждый слой обслуживает вышележащий слой, выполняя для него некоторый набор функций, которые образуют межслойный интерфейс. На основе этих функций следующий верхний по иерархии слой строит свои функции – более сложные и более мощные и т.д. Такая организация системы существенно упрощает ее разработку, т.к. позволяет сначала "сверху вниз" определить функции слоев и межслойные интерфейсы, а при детальной реализации, двигаясь "снизу вверх", – наращивать мощность функции слоев. Кроме того, модули каждого слоя можно изменять без необходимости изменений в других слоях (но не меняя межслойных интерфейсов!).

## 1. Что представляет собой ОС?

**Операционная система** — комплекс программ, обеспечивающий управление аппаратными средствами компьютера, организующий работу с файлами и выполнение прикладных программ, осуществляющий ввод и вывод данных.

Общими словами, операционная система — это первый и основной набор программ, загружающийся в компьютер. Помимо вышеуказанных функций ОС может осуществлять и другие, например предоставление общего пользовательского интерфейса.

Сегодня наиболее известными операционными системами являются ОС семейства Microsoft Windows и UNIX-подобные системы.

## 2. Что понимаем под состояниями *user* и *supervisor*?

Режим супервизора — привилегированный режим работы процессора, как правило используемый для выполнения ядра операционной системы.

В данном режиме работы процессора доступны привилегированные операции, как то операции ввода-вывода к периферийным устройствам, изменение параметров защиты памяти, настроек виртуальной памяти, системных параметров и прочих параметров конфигурации.

Как правило, в режиме супервизора или вообще не действуют ограничения защиты памяти или же они могут быть произвольным образом изменены, поэтому код, работающий в данном режиме, как правило, имеет полный доступ ко всем системным ресурсам (адресное пространство, регистры конфигурации процессора и т. п.). Во многих типах процессоров это наиболее привилегированный режим из всех доступных режимов.

Режим юзера - пользовательский режим (непривилегированный режим работы процессора, в котором исполняется код прикладных программ, и предоставляющий исполняемому в нем процессу доступ к системным ресурсам только посредством вызова системных сервисов)

## 3. Назовите 4 функции которые должна выполнить компонента управления информацией

- создание/удаление файлов
- открытие/закрытие файлов,
- чтение/запись файлов
- обработка конфликтных ситуаций, связанных с одновременным доступом нескольких процессов к одному файлу

## 4. Назовите простейшие функции 5-ти уровней иерархической машины

- *управление процессорами*
- *управление памятью* (выделение и освобождение памяти)
- *управление процессами* (создание и уничтожение процесса, передача сообщений между процессами, возобновление/приостановление работы процесса)
- *управление устройствами* (наблюдение за состоянием всех устройств, координация операций ввода/вывода)
- *управление информацией* (создание/удаление файлов, открытие/закрытие файлов, чтение/запись файлов)

## 5. Что означает маскированное прерывание?

**Прерывание** (англ. *interrupt*) — сигнал, сообщающий процессору о наступлении какого-либо события. При этом выполнение текущей последовательности команд приостанавливается, и управление передаётся обработчику прерывания, который реагирует на событие и обслуживает его, после чего возвращает управление в прерванный код

Внешние прерывания в зависимости от возможности запрета делятся на: маскируемые и немаскируемые.

**маскируемые** — прерывания, которые можно запрещать установкой соответствующих битов в регистре маскирования прерываний (в x86-процессорах — сбросом флага IF в регистре флагов);

## 6. Назовите 4 главные х-ки согласно которым описывается стратегия выделения памяти

Память является важнейшим ресурсом, требующим тщательного управления со стороны мультипрограммной операционной системы. Распределению подлежит вся оперативная память, не занятая операционной системой. Обычно ОС располагается в самых младших адресах, однако может занимать и самые старшие адреса. Функциями ОС по управлению памятью являются:

- отслеживание свободной и занятой памяти,
- выделение памяти процессам и освобождение памяти при завершении процессов, -вытеснение процессов из оперативной памяти на диск, когда размеры основной памяти не достаточны для размещения в ней всех процессов,
- возвращение их в оперативную память, когда в ней освобождается место, а также настройка

адресов программы на конкретную область физической памяти.

#### **7. Назовите достоинства и недостатки метода «единственный резидентный пользователь»**

Понятие существования или жизни объекта может быть распространено на пути доступа, что означает, что время их существования (время, отделяющее создание разрушения).

- 1) Внутренние объекты: это инструкции, которые составляют текст процедуры. Их время жизни совпадает с временем жизни процедуры.
- 2) Локальные объекты: это переменные, объявленные в процедуре. В случае рекурсивного вызова новая копия каждого локального объекта создается при каждом вызове, а его идентификатор обозначает последний созданный (оставшийся остается недоступным до возврата на соответствующий уровень).
- 3) Остаточные объекты и глобальные объекты: это объекты, которые уже существовали по призыву процедуры и которые выживут при возврате; их время жизни - это либо процесс (остаточные объекты), либо его процедура, которая охватывает данную процедуру (глобальные объекты).
- 4) Внешние объекты: объекты, созданные и сохраненные независимо от рассматриваемой процедуры и процесса (другие процедуры, файлы и т. Д.). Их продолжительность жизни не зависит от процедуры или процесса; они могут быть созданы или уничтожены динамически во время выполнения процедуры.
- 5) Параметры: формальные параметры - это идентификаторы, используемые внутри процедуры и связанные только во время разговора.

## Билет №3

### 1. Назовите 4 причины необходимости изучения ОС

Есть несколько причин, необходимость изучения операционных систем, наиболее важным из которых:

- для использования hardware для специального назначения, возможно, потребуется создать собственную операционную систему или внести изменения в существующую;
- Правильный выбор операционной системы и конкретного варианта может зависеть от жизнеспособности и эффективности расчетной системы;
- идеально подходит пользователю взаимодействовать с операционной системой, зная все её тонкости, потому что операционная система является посредником между компьютером и пользователем;
- Многие методы и понятия, используемые в области ОС, могут быть успешно реализованы в других областях.

### 2. Назовите 4 типа ресурсов

Ресурсом является любой объект, который может распределяться внутри системы. Ресурсы могут быть разделяемыми, когда несколько процессов могут их использовать одновременно (в один и тот же момент времени) или параллельно (в течение некоторого интервала времени процессы используют ресурс попеременно), а могут быть и неделимыми.

Существует 4 типа ресурсов:

- процессорное время,
- память,
- каналы ввода/вывода
- периферийные устройства.

### 3. Нарисуйте и прокомментируйте схему управления технологическим процессом

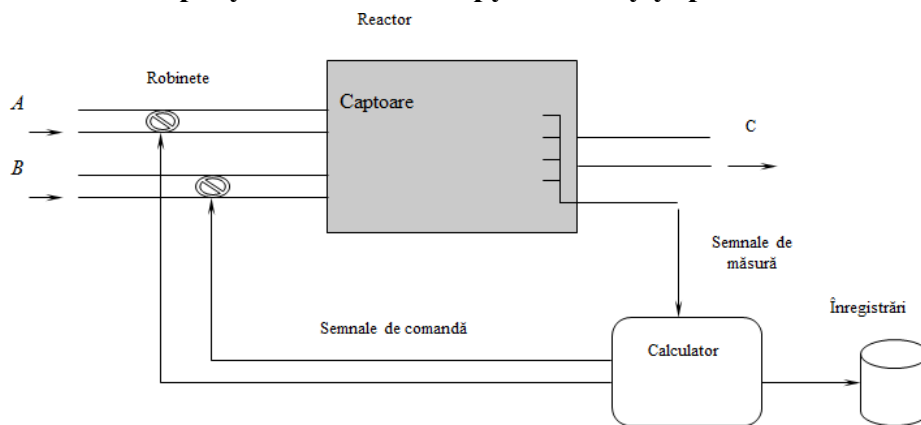


Fig.1.6. Schema unui proces chimic

Производственный процесс управляется компьютером, который выполняет следующие функции:

- **Регулировка** Для хорошего текущих эксплуатационных параметров производственного процесса (температура, давление, концентрация и т.д.) должны быть в пределах предварительно заданных значений Это будет зависеть приток товара А или параметров Б. Компьютер принимает эти измерения и, в зависимости от алгоритма управления, ввода клапанами
- **Регистрация** Результаты измерения регулярно регистрируются; их значения отображаются на приборной панели и переписал в файле ("Борт-журнал") для дальнейшей обработки (статистики)
- **Безопасность** Если один из параметров, измеренных превышает predetermined критический реактор должен быть немедленно прекращена

Этот режим вводит некоторые ограничения:

1. Измерения происходят периодически;  $T$  значение интервала времени между двумя последовательными измерениями (период дискретизации) и  $t$  - общее время обработки данных на ЭВМ (измерение фактических наблюдаемые сигналы, записи, вычисления управляющих сигналов и эксплуатации клапанов) Система будет работать, только если отношения уважать  $t \leq T$
2. Система безопасности имеет приоритет Превышение критических значений должен быть обнаружен в любой момент и лечения этих травм будет приостановить все операции в прогресс.

### 4. Контекстом называется...

Контекстом процесса является его состояние, определяемое текстом, значениями глобальных переменных пользователя и информационными структурами, значениями используемых машинных регистров, значениями, хранимыми в позиции таблицы процессов и в адресном пространстве задачи, а также содержимым стеков задачи и ядра, относящихся к данному процессу. Текст операций системы и ее глобальные информационные структуры совместно используются всеми процессами, но не являются

составной частью контекста процесса.

## 5. Описать метод активного ожидания

Чтобы иметь дело с активным ожиданием обновления нескольких процессов и консультаций с общими переменными, некоторые машины имеют инструкцию, которая неделимым образом выполняет консультации и обновление местоположения памяти. Эта команда, часто называемая Test And Set (tas), используется в многопроцессорных системах (в монопроцессорных системах маскировка прерываний является достаточной для обеспечения взаимного исключения).

Либо адрес рассматриваемой ячейки памяти, либо замка, и R - регистр процессора. По соглашению, если замок находится в 0, критический раздел свободен, и если он 1 - он занят. Действие Test and Set описано ниже (Mp [m] обозначает местоположение памяти с адресом m):

```
tas R, m : <blocare acces la Mp[m]>  
          R:=Mp[m]  
          Mp[m]:=1  
          <eliberare acces la Mp[m]>
```

## 6. Как можно ограничить время исполнения программы?

Ограничение времени выполнения программы может быть выполнено с помощью отклонений, из-за которых выдается сообщение об ошибке, чтобы остановить текущую программу и перейти к следующей программе.

## 7. Опишите алгоритм записи в методе ВВ в буферизированной памяти

Значительное различие между скоростью центрального блока и периферийными устройствами основано на «буферизации» ввода, но это также буфер между периферийным устройством и программистом. Все это в объеме, но центральный блок не работает, но он не работает. Программа опроса была обновлена информацией в буферной зоне или в буферной зоне, и буфер был назначен исходному или назначенный параллельный обмен данными с периферийным устройством. Дерегулирование центрального блока увеличивается с увеличением пропускной способности буферной зоны. Иногда вам не нужно создавать резервные копии основной памяти, буфера на фиксированном диске.

### 1) Чтение

Они представляют свои собственные процедуры, которые являются предметом разбирательства. Процедура, использование супервизора, имя выбранного сообщения в буферной зоне. Прочтите это полностью или частично, если буфер не сохранен, файл не будет считан. В противном случае активация подписчиков абонента находится в состоянии ожидания. В случае непереводимых буферов учетные записи вводятся с входными периферийными устройствами, но данные не используются.

## 8. Специфицировать функцию доступа открытие файла

Файл f может быть открыт для чтения или записи. Некоторые организации разрешают писать с определенной записи. Мы считаем, что запись делается в конце файла и открыв для записи инициализируется файл в <vid>

Открытие достигается за счет операции:

```
deschide(mod):  
    if mod=citire then  
        f.rest:=<şirul înregistărilor fişierului>
```

else

```
        f:=<vid>
```

endif;

```
    f.mod:=mod;
```

```
    avansare
```

где функция avansare определяется следующим

образом: if f.rest=vid then

```
    f.sfârşit:=true
```

else

```
    f.p    rimul:=<pointer    pe
```

```
    primul(f.rest)>;    f.rest:=f.rest-
```

```
    primul(f.rest)
```

endif

## 9. Что называем независимые процессы?

Если два взаимосвязанных процесса при развитии используют совместно некоторые ресурсы, но информационно между собой не связаны, т. е. не обмениваются информацией, то такие процессы называют информационно- независимыми. Связь между такими процессами может быть либо

функциональная, либо пространственно- временная. При наличии информационных связей между двумя процессами их называют взаимодействующими, причем схемы, а следовательно, и механизмы установления таких связей могут быть различными. Особенность, во- первых, обусловлена динамикой процессов (т. е. являются ли взаимодействующие процессы последовательными, параллельными или комбинированными); во-вторых, выбранным способом связи (явным, с помощью явного обмена сообщениями между процессами, или неявным, с помощью разделяемых структур данных). Когда необходимо подчеркнуть связь между взаимосвязанными процессами по ресурсам, их называют конкурирующими.

#### 10.Что означает *событие с памятью (eveniment memorizat)*?

Событие с памятью является переменной, которая может принимать два значения : non\_sosîт и sosîт , начальное значение non\_sosîт. С данным событием возможно провести две операции, которые являются неделимыми действиями :

e : = <значение > - немедленное присвоение значения ожидание<e>.

Операция ожидание (е осуществляется с помощью процесса p имеет следующую спецификацию :

```
if e = non_sosîт then
    starea(p) := blocat    -- p este trecut în “aşteptarea
    lui e” endif
```

Когда е принимает значение sosîт , все процессы которые находятся в режиме е переходят в активное состояние..

Пример

```
var e : eveniment memorizat;
procesul p      procesul
q
scriere(a);      <debut q>;
e:=sosîт;        aşteptare(e);
<continua_re_p> citire(a) ◀
```

## Билет 4

### 1.Что представляет собой ОС?

ОС – это набор взаимосвязанных программ, предназначенных для управления ресурсами компьютера и организации взаимодействия с пользователем.

### 2.В чем состоит главное назначение ОС с точки зрения использования ресурсов?

С точки зрения использования ресурсов, главное назначение ОС – это предоставлять пользовательским программам доступ к ним, а так же решать конфликтные ситуации, которые могут возникать при использовании общих ресурсов, таких как: оперативная память, файлы, устройства ввода-вывода

### 3.Назовите простейшие функции 5-ти уровней иерархической машины.

- управление процессорами
- управление памятью (выделение и освобождение памяти)
- управление процессами (создание и уничтожение процесса, передача сообщений между процессами, возобновление/приостановление работы процесса)
- управление устройствами (наблюдение за состоянием всех устройств, координация операций ввода/вывода)
- управление информацией (создание/удаление файлов, открытие/закрытие файлов, чтение/запись файлов)

### 4.Назовите основные требования к ОС внесенные СУ технологическим процессом.

- *Безопасность.* Превышение критических значений должно быть обнаружено немедленно.
- *Учёт реального времени*
- *Выполнение периодических измерений.* При этом время обработки данных, должно быть  $\leq$  периода времени между двумя последовательными измерениями.

### 5.Контекст активности состоит из ...

... контекста процессора (программируемые и внутренние регистры) и контекста памяти (сегмент процедуры и сегмент данных)

### 6.Назовите операции, которые имеют место при коммутации контекста.

- сохранение и восстановление контекста при вызове и возвращении из процедуры
- передача параметров между вызывающей и вызванной процедурами
- администрирование рабочей зоны процедуры, с возможностью рекурсивных вызовов

### 7.Как снимаются данные измерения?

Одним из примеров использования прерываний является сбор данных измерения. Например, компьютер должен периодически принимать данные некоторых измерений, связанных с промышленным процессом. Период часов - 5  $\mu$ s. Данные должны быть собраны каждые 100 мс (период измерения). Т.к. время необходимое для сбора данных намного меньше периода измерения, то компьютер может выполнять другие процессы в фоне.

### 8.Что понимаем под понятием уровень наблюдения?

Под понятием уровень наблюдения понимаем то, с какой точки зрения мы рассматриваем тот или иной феномен. Например операция сложения чисел на уровне инструкций высокого языка программирования это  $a + b$

### 9.Что собой представляет монитор и каковы его основные функции?

Монитор – это множество переменных состояния и процедур, которые их используют. Некоторые из этих процедур, внешние, доступны пользователям монитора.

Монитор используется для синхронизации процессов согласно условию. Условие блокировки/разблокировки выражается как функция переменных состояния, которыми управляет монитор.

## Билет 5

### 1. Что представляют собой процессоры с точки зрения ОС?

С точки зрения ОС, процессор представляет собой общий ресурс, необходимый для выполнения команд.

### 2. Назовите 4 функции подсистемы управления информацией

- создание/удаление файлов
- открытие/закрытие файлов,
- чтение/запись файлов
- обработка конфликтных ситуаций, связанных с одновременным доступом нескольких процессов к одному

файлу

### 3. Приведите пример организации ЭВМ

Например, ПК. Самая простая конфигурация ПК это: центральный процессор, системная плата, основная и второстепенная память, дисплей, клавиатура, мышь.

Пользователь ПК требует выполнения 2 основных функций:

- манипулирование файлами и множеством информации, хранение этих файлов в второстепенной памяти, а также передача информации между файлами и устройствами ввода/вывода
- выполнение программ

### 4. Назовите основные функции ОС в случае управления технологическим процессом

- управление внешними устройствами
- учёт реального времени
- реакция на внешние события
- управление информацией (например файлом журнала)

### 5. Как осуществляется переход от одной активности к другой?

Переход от одной активности к другой осуществляется с помощью специальных инструкций: **вызов** и **возвращение** из процедуры, которые реализуют коммутацию контекста.

### 6. Опишите две наиболее часто применяемых схемы коммутации контекста

Две схемы коммутации контекста:

1. **Сохранение в установленные ячейки памяти.** Каждой причине коммутации контекста соответствуют две ячейки памяти:  $csp\_vechi[i]$  и  $csp\_nou[i]$  ( $i$  – номер причины).

$Mp[csp\_vechi[i]] := \langle \text{cuvânt de stare program} \rangle$   
 $\langle \text{cuvânt de stare program} \rangle := Mp[csp\_nou[i]]$

2. Сохранение в стеке.

$ptr := ptr + 1$   
 $stiva[ptr] := \langle \text{cuvânt de stare program} \rangle$   
 $\langle \text{cuvânt de stare program} \rangle := Mp[csp\_nou[i]]$

При восстановлении слова состояния:

$\langle \text{cuvânt de stare program} \rangle$   
 $:= stiva[ptr] \quad ptr := ptr - 1$

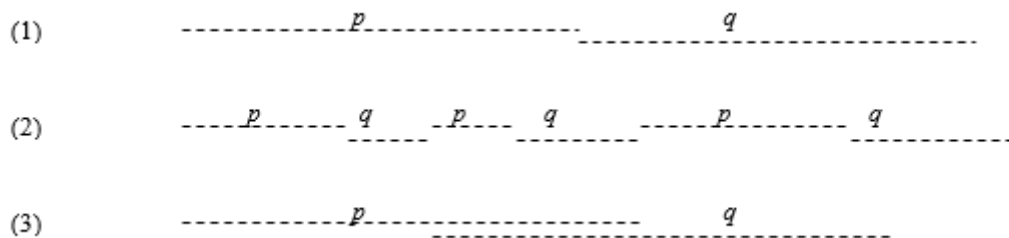
### 7. Опишите алгоритм управления заданиями в режиме разделения времени

В режиме разделения времени, всем запущенным процессам попеременно выделяется квант процессорного времени. При каждом переходе на другой процесс, контекст текущего процесса должен быть сохранён, чтобы впоследствии продолжить его исполнение. В общем случае следует сохранить только контекст процессора. Однако, если основная память может содержать в данный момент времени сегмент процедуры и данные только одного процесса, тогда эти сегменты тоже должны быть сохранены.

### 8. Приведите примеры, объясняющие понятия реального и псевдо-параллелизма.

Пусть, имеем два процесса: P и Q. Их совместное исполнение может происходить по разному.





Вышеупомянутые схемы можно прокомментировать следующим образом:

- Схема 1: сначала выполняется весь процесс  $p$ , затем весь процесс  $q$ ,
- Схема 2: Выполнение командных строк процесса  $p$  в альтернативном варианте с инструкциями строки  $q$ , и так до завершения обоих процессов,
- схема 3: одновременное выполнение процессов  $p$  и  $q$ ; В этом случае необходимы два процессора.

Анализируя схемы уровня наблюдения, к которым, по соглашению, полное выполнение каждой из программ  $P$  и  $Q$  представляет собой одно действие, мы говорим, что:

а) схема типа 1 представляет собой последовательное выполнение  $p$  и  $q$ . Он характеризуется отношениями:

$\text{end}(q) < \text{начало}(p)$  или  $\text{окончание}(p) < \text{начало}(q)$

б) Схемы типа 2 или 3 представляют собой схемы параллельного выполнения. Они характеризуются

$\text{end}(p) > \text{start}(q)$  или  $\text{end}(q) > \text{start}(p)$ .

Анализируя схемы уровня инструкций (базовый уровень), мы можем различать схемы 2 и 3. На схеме 2, поскольку в данный момент только один процессор, может выполняться только одно действие, в отличие от схемы 3. Будет сказано, что на схеме 3 существует реальный параллелизм, а на схеме 2 - псевдопараллельность. Реальный параллелизм требует двух разных процессоров

## 9. Что представляет собой условие монитора?

Блокировка и разблокировка процессов выполняется в процедурах монитора с помощью условий. Условие объявляется как переменная, но не имеет значения.

## Билет 6

### 1. Что называется ядром вычислительного комплекса?

Ядром вычислительного комплекса является **операционная система**, которая состоит из программных модулей, управляющих техническими ресурсами.

### 2. Представить графически и комментировать элементарные состояния процесса



1. Процесс заблокирован в ожидании ввода
2. Диспетчер выбрал другой процесс
3. Диспетчер выбрал данный процесс
4. Входные данные стали доступны

Процесс может находиться в 3-х основных состояниях: *действие*, *готовность* и *блокировка*.

Стрелками показаны переходы между этими состояниями.

Действие соответствует тому, что процессор исполняет процесс. Готовность – процесс временно приостановлен, чтобы позволить выполниться другому процессу. Блокировка – процесс не может быть возобновлён, пока не произойдёт некое внешнее событие.

### 3. Какие основные характеристики ПК интересуют простого пользователя?

Простого пользователя интересуют такие характеристики ПК, как: скорость работы, способность работать с несколькими приложениями одновременно, размерность основной и второстепенной памяти, а так же наличие возможности подключения различных периферийных устройств.

### 4. Назовите основные х-ки транзакционных систем

- система управляет набором информации или БД больших объёмов
- над этой информацией может быть выполнен определенный набор операций или транзакций
- система имеет много точек доступа и большое число транзакций, которые могут выполняться одновременно.

### 5. Назовите этапы выполняемые при вызове и возврате из процедуры

Например, вызывающая процедура Р вызывает процедуру Q. При этом выполняются след. этапы:

- (1) Подготовка параметров, передаваемых от процедуры Р к Q
- (2) Частичное сохранение контекста Р, необходимого при возвращении к этой процедуре
- (3) Замена контекста процедуры Р контекстом процедуры Q При возвращении из Q:
  - (1) Подготовка результатов, передаваемых процедурой Q в Р
  - (2) Восстановление контекста процедуры Р, сохранённого до её вызова

### 6. Как можно разделить прерывания на три категории?

В зависимости от источника возникновения сигнала прерывания делятся на:

- **асинхронные**, или внешние (аппаратные) — события, которые исходят от внешних источников (например, периферийных устройств) и могут произойти в любой произвольный момент
- **синхронные**, или внутренние — события в самом процессоре как результат нарушения каких-то условий при исполнении машинного кода, например деление на ноль или переполнение стека
- **программные** (частный случай внутреннего прерывания) — инициируются исполнением специальной инструкции в коде программы. Программные прерывания как правило используются для обращения к функциям встроенного программного обеспечения (firmware), драйверов и операционной системы.

### 7. Что представляет собой канал, контроллер, п.у. (периферийное устройство)?

**Канал** – это процессор, специализированный на операциях В/В. Его набор инструкций позволяет управлять контроллерами и периферийными устройствами, которые к нему подключены.

**Контроллер** – это устройство управления, адаптированное под определённый тип периферийных устройств. Его главное назначение – это позволить подключение многих периферийных устройств одинакового типа.

**Периферийное устройство** – это устройство, которое позволяет вводить информацию в компьютер или выводить ее из него.

**8. Что называется критическим ресурсом?**

Пусть, имеем множество процессов, контексты которых имеют общий объект, который может быть использован в данный момент времени только одним процессом. Тогда говорим, что этот объект является **критическим ресурсом** для данных процессов.

**9. Специфицируйте три примитивы условия монитора**

Блокировка и разблокировка процессов выполняется в процедурах монитора с помощью условий. Условие объявляется как переменная, но не имеет значения. Условие “с” может быть использовано только посредством 3-х операций (примитивов):

c.așteptare : блокирует процесс p и ставит его в “ожидание с”

c.vid : функция с булевым значением (true, если не существует не одного процесса в ожидании с, иначе false)

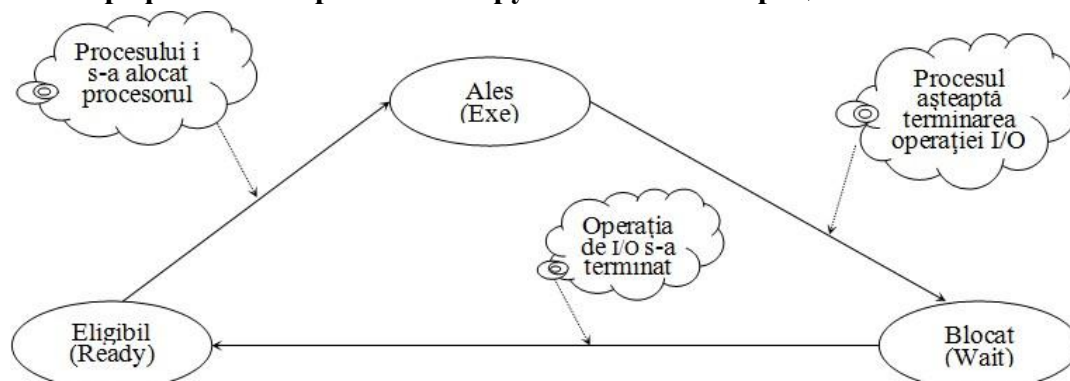
c.s emnalizare : if non c.vid then <разблокировка процессов находящихся в ожидании с> endif

## Билет 7

### 1. Какие устройства вычислительного комплекса называют периферийными?

Периферийными (или внешними) называют все устройства (вместе с их интерфейсами), подключаемые к компьютеру, не входящие в его логические блоки (т.е. процессоры) или память, которые позволяют вводить информацию в компьютер и выводить её из него.

### 2. Представьте графически и прокомментируйте состояния процесса



Три базовыми состояниями процесса являются: "исполнение" (exe) - процессу выделен процессор, и он выполняется; "готовность" (ready) - процессу доступны все необходимые для исполнения ресурсы, он ожидает только выделения процессора; и "ожидание" (wait) - процесс блокируется в ожидании выполнения какого-либо события. ОС переводит процесс из одного состояния в другое.

### 3. Какие основные вопросы интересуют профессионала когда он знакомится с ЭВМ?

В какой операционной системе запущен / принят этот компьютер?

Каковы возможности, что может предложить операционная система?

f Какие приложения могут быть установлены на этом компьютере?

### 4. Назовите основные свойства ОС транзакционного типа

Система транзакционного типа обладает следующими свойствами:

- система управляет набором или базами данных;
- над этой информацией можно выполнять определенный набор заранее заданных операций или интерактивных транзакций;
- система обладает большим количеством точек доступа, одновременно может иметь место большое число транзакций может.

Обязательными свойствами такой системы являются доступность и надежность, а также устойчивость к дефектам. Также важным является большое количество параллельных процессов (и иногда - географическое расположение компонентов системы).

### 5. Перечислите функции механизма последовательного исполнения программ

Механизм последовательного исполнения выполняет следующие функции:

- сохранение и восстановление контекста при вызове и возврате;
- передача параметров между вызывающими и вызываемыми процедурами;
- администрирование рабочей зоны каждой процедуры, с разрешенными рекурсивными вызовами.

### 6. Что представляет собой обработчик прерываний?

Обработчик прерываний (interrupt handler) - это специальная процедура, вызываемая по прерыванию для выполнения его обработки. Это заставляет процессор приостановить выполнение текущей задачи. Обработчик прерываний выполняется в контексте, отличном от контекста прерванной программы, что связано с режимом обработки, защиты, доступной информации и т.д.

### 7. Что называется, операцией ввода/вывода и что она означает?

Под "вводом/выводом" понимается любая передача данных из или в ядро процессора. Операции ввода/вывода означают:

- передачу информации между разными иерархическими уровнями памяти;
- передачу информации из или во внешнюю среду (локальные или удаленные периферийные устройства, другие компьютеры и т.п.).

### 8. Каковы основные причины введения понятия "виртуальный ресурс"?

Понятие виртуального ресурса подразумевает, что для каждого критичного физического ресурса

создается столько виртуальных копий, сколько на данный момент есть процессов, нуждающихся в данном ресурсе. Использование виртуальных ресурсов позволяет создать видимость параллельного выполнения процессов. Использование виртуальных ресурсов также позволяет нам обрабатывать проблему взаимного исключения при доступе к физическим ресурсам в рамках механизмов выделения ресурсов.

#### **9. Назовите основные проблемы, связанные с синхронизацией**

Типичные проблемы, связанные с имплементацией синхронизации:

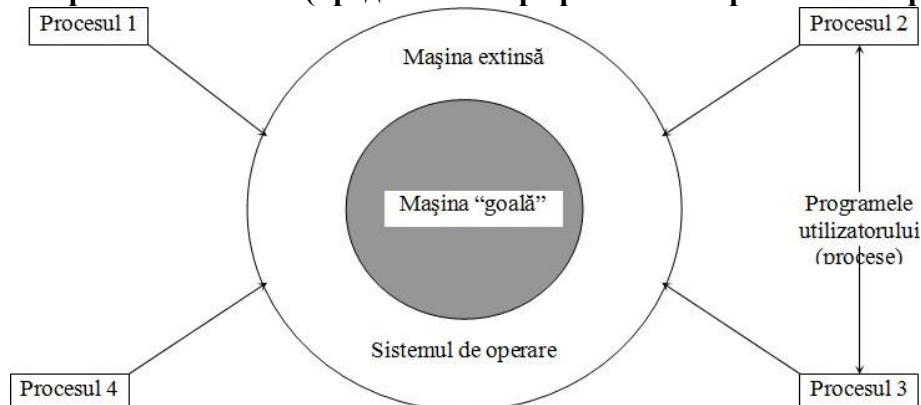
- попытка доступа к общему ресурсу несколькими процессами, разделяющими его;
- коммуникация между процессами;
- управление периферийными устройствами и буферами ввода/вывода;
- синхронизация во времени;

## Билет 8

### 1. Определите понятие “задача”

Под задачей (task) понимается набор действий, необходимых для выполнения какой-то работы. Задача может состоять из нескольких шагов. Шаги задачи являются рабочими единицами, которые выполняются последовательно. Получая задачу от пользователя ОС может создать несколько процессов.

### 2. Что такое расширенная машина (представить графически и прокомментировать)?



Расширенная машина представляет собой совокупность компьютера с его стандартным набором инструкций и операционной системы, предоставляющей дополнительный набор инструкций. Машина на схеме изображает ядро ОС, которое выполняется “железом”, а программы пользователя выполняются расширенной машиной.

### 3. Что представляет собой регистр флагов условий?

Регистр флагов отражает текущее состояние процессора. Он содержит группу флагов состояния, управляющий флаг и группу системных флагов. Значение некоторых флагов в регистре флагов можно изменять напрямую, с помощью специальных инструкций (например, CLD для сброса флага направления), но нет инструкций, которые позволяют обратиться (проверить или изменить) к регистру флагов как к обычному регистру.

Флаг или код условий - это последовательность битов, устанавливаемых или сбрасываемых процессором в зависимости от результата выполненных операций.

### 4. Каково основное назначение ОС с разделением времени?

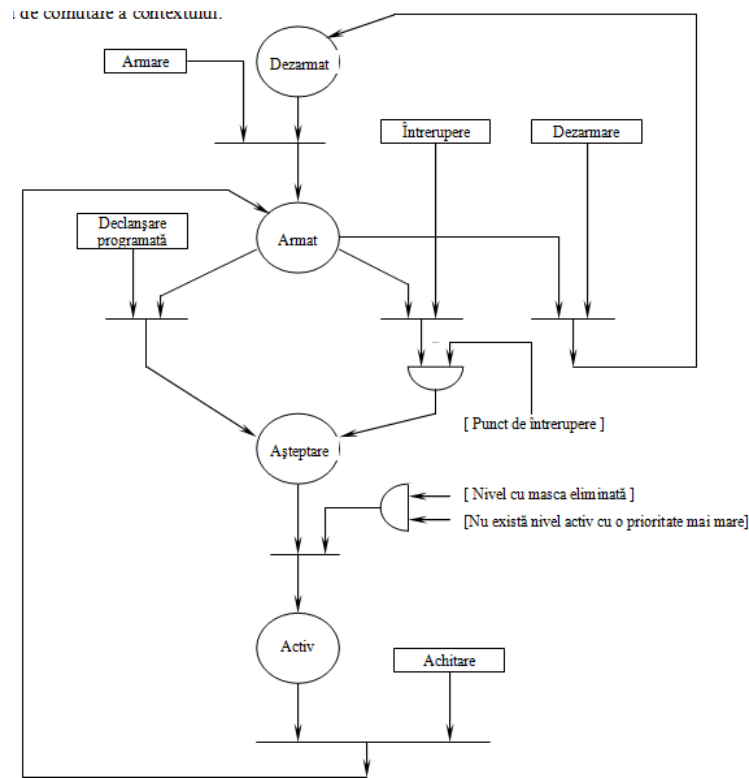
ОС с разделением времени должна гарантировать каждому пользователю приемлемое время ответа на его запрос. Этого можно добиться последовательным выделением времени процессора программам пользователей на очень короткие промежутки времени (кванты).

### 5. Какие структуры данных используются для процедур? Что есть “среда выполнения”?

При последовательном выполнении программ используется стек вызовов - LIFO-стек, хранящий информацию для возврата управления из процедур в программу (или процедуру, при вложенных или рекурсивных вызовах) или для возврата в программу из обработчика прерываний.

При каждом вызове процедуры на вершине стека вызовов создается структура данных, называемая “средой выполнения”, которая будет удалена после соответствующего возврата.

### 6. Опишите граф переходов для некоторого уровня прерываний



Каждому уровню соответствует отдельная программа, обрабатывающая заданное прерывание, которая активируется автоматически механизмом коммутации контекста.

Для некоторых уровней эффект поступления сигнала прерывания можно скрыть. В этом случае уровень “запрещается” (dezarmat). Такой уровень можно “разрешить” (armat), т.е. заново ввести в работу.

Для некоторых компьютеров существует инструкция “запрограммированного запуска” (declansare programata), которая позволяет модифицировать индикатора, соответствующего с каким-либо уровнем прерывания, из какой-либо программы.

В случае, когда все необходимые для прерывания условия выполняются, уровень считается “активным”. Это состояние аналогично выполнению обработчика прерываний. выход из активного состояния производится через “выключение”(achitare) прерывания.

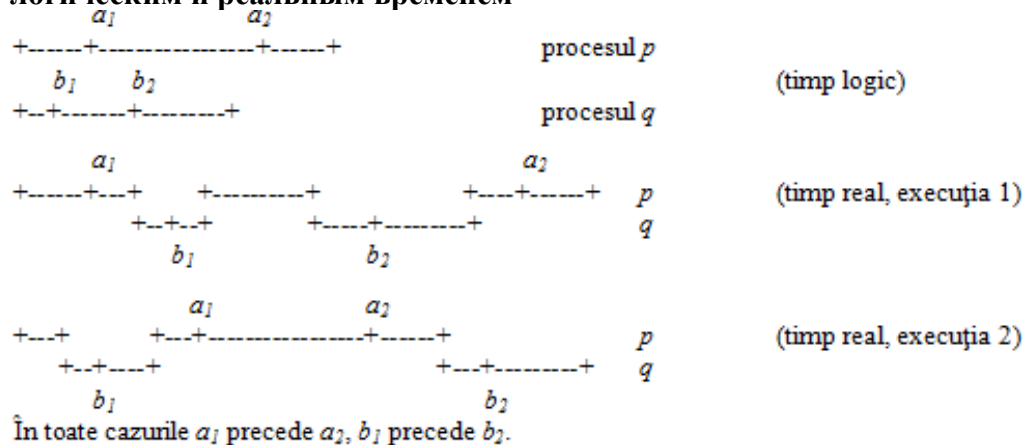
### 7. Опишите схему адресации периферийных устройств

Разным элементам на разных уровнях даются адреса, с помощью которых можно определить:

- каналы, связанные с памятью;
- контроллеры, прикрепленные к каждому каналу;
- периферийные устройства, прикрепленные к каждому контроллеру.

Контроллер может быть соединен с несколькими каналами, а периферийное устройство - с несколькими контроллерами, которые обязательно должны быть одного типа. Таким образом периферийное устройство обладает несколькими путями доступа к памяти (только один может быть активен в любой момент времени) и может иметь несколько адресов. Такая организация позволяет улучшить производительность и доступность системы, уменьшает риск недоступности периферийного устройства и выхода из строя канала или контроллера.

### 8. Связь между логическим и реальным временем



Логическое время - время, необходимое для определения положения в логической схеме событий.

Если допустить, что процессы протекают в логическом времени, относительная скорость процессов теряет свое значение - это упрощение вводится понятием виртуальных ресурсов, которые позволяют игнорировать механизмы выделения. При рассмотрении синхронизации во времени реальное время является не только относительным способом определения положения, но точным образом измеряет длительности процесса.

#### **9. Что такое “делимый ресурс”**

Предположим, что имеется ресурс с ограниченным количеством экземпляров. Процесс может использовать и затем освободить какое-то количество этих экземпляров, другими словами экземпляр выделяется процессу. С точки зрения использующих их процессов все эти единицы ресурсов равны между собой. В таком случае говорят, что ресурс является делимым. Примерами таких ресурсов являются буферы памяти, секторы а гибких дисках и т.п.

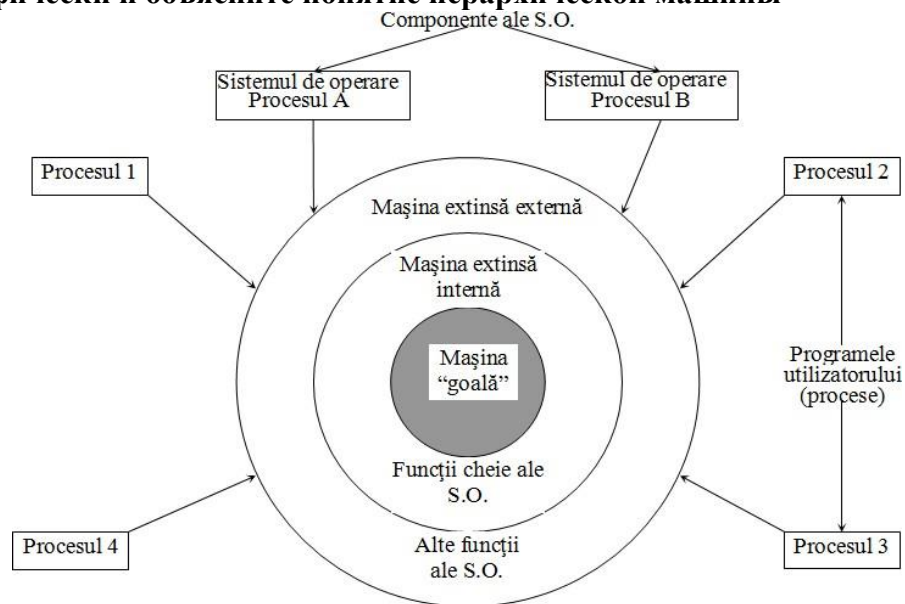


## Билет 9

### 1. Назначение ОС с точки зрения адресного пространства

Совокупность программ и данных, доступных процессу в момент выполнения, формирует адресное пространство. Одним из назначений ОС является необходимость удостовериться в том, что адресное пространство некоего процесса проецируется в физическую память. Для этого используются как технические (сегментация памяти или постраничное разделение), так и программные средства.

### 2. Изобразите графически и объясните понятие иерархической машины



Иерархическая машина использует принцип расширенной машины, но на двух уровнях. Первый уровень представляет собой ключевые функции, используемые большинством модулей системы (внутренняя расширенная машина), второй - некоторые модули могут выполняться в рамках внешней расширенной машины, аналогично процессам пользователя.

### 3. Что такое PSW?

Слово статуса программы/процессора (program status word) - контрольный регистр, который выполняет функции статусного регистра или программного счетчика. PSW содержит информацию о состоянии процессора:

- состояние выполнения (процессор может быть в активном состоянии или в состоянии ожидания);
- режим работы (master-slave);
- маску прерываний;

PSW также содержит информацию о контексте памяти и соответствующих правах доступа. Кроме того он содержит информацию о выполнении текущего процесса: код условия, счетчик следующей операции и т.д.

### 4. Какие проблемы связаны с принципом разделения времени?

При обслуживании нескольких пользователей система использует время, которое каждый пользователь тратит на обдумывание своих действий, на обработку запросов, поступающих одновременно от других пользователей. Если предположить, что программы всех пользователей находятся в памяти, время коммуникации между двумя программами сводится ко времени снятия процессора с одной задачи и выделения для другой. Несмотря на то, что этим временем можно пренебречь, становится необходимой поддержка многозадачности.

На деле всё осложняется невозможностью хранения программ всех пользователей в памяти одновременно. Возникает вероятность отсутствия нужной программы в памяти при попытке обратиться к ней со стороны процессора. ОС должна минимизировать эту вероятность.

### 5. Назовите три операции, выполняемые при вызове процедуры

При каждом вызове процедуры на вершине стека вызовов создается структура данных, называемая "средой выполнения", которая будет удалена после соответствующего возврата.

Сначала среде выполнения выделяется зона в стеке (размер известен заранее), затем сохраняется информация возврата, сортируются параметры, и вызванная процедура выполняет ветвление(?).

### 6. Что представляет собой "отклонение"?

Отклонение (trap) указывает на аномальное поведение в ходе выполнения инструкции, которое препятствует успешному выполнению данной инструкции. Причинами такого поведения могут быть:

- неправильные данные (например, деление на ноль);
- неопределенность в результате выполнения запрещенной операции (например, нарушение режима защиты памяти);
- невыполнимая инструкция (запрещенный код операции, адрес за пределами действительной памяти и т.п.).

## **7. Опишите алгоритм синхронного ввода/вывода, его преимущества и недостатки**

Драйвер - это программа, контролирующая базовую функциональность периферийного устройства. Как правило, пользователь не взаимодействует с драйверами напрямую и вызывает функции ввода/вывода через сервисы высокого уровня. В этом случае между обработкой и передачей информации нет никакого параллелизма, и процессор остается занятым на протяжении всего периода передачи данных.

Предположим, что нам нужно вывести последовательность символов и массива с заданной длиной. Каждый последующий символ может быть извлечен только если до него был извлечен стоящий перед ним символ. Для этого статус передачи проверяется в цикле на завершенность.

Синхронный ввод/вывод используется для простейших микропроцессоров или в тех случаях, когда процессор не может использоваться другим процессом в момент передачи данных.

## **8. Приведите примеры где возникает проблема взаимного исключения**

Предположим, что есть несколько процессов, контексты которых имеют общий объект. В любой момент времени этот объект может использовать только один процесс. В этом случае говорится, что объект является критическим ресурсом, или что процессы друг друга взаимно исключают в попытке использования ресурса.

Например, допустим, что обработка прерывания, запущенная окончанием физической передачи данных (которая производилась параллельно с выполнением примитива “чтение” процессором), отвечает на прерывание в ходе выполнения этого примитива. Прерывание установит какие-то переменные в определенные значения, отличные от ожидаемых программой. Это может привести к заикливанию программы (например, установке переменной из true в false предотвращает запланированный запуск периферийного устройства, и в результате вызывающий процесс остается в режиме ожидания).

Правильность общих переменных невозможно гарантировать если разрешить прерывания операций ввода/вывода в любой момент времени. В таких случаях эти прерывания необходимо маскировать до выполнения вызова инструкции чтения/записи.

## **9. Опишите модель “читатель-писатель”**

Представим себе файл, манипулируемый двумя процессами, принадлежащими разным классам: “читателям”, которые не могут изменять содержимое файла, и “писателям”, которые могут изменять файл. Для гарантирования целостности файла необходимо убедиться в том, что в момент чтения файла количество писателей равно 0, а количество читателей больше либо равно 0; также в момент записи файла количество читателей должно быть равно 0, при том, что писатель должен быть только 1.

Этого можно добиться, расставив приоритеты для читателей и писателей. Затем достаточно проверять уровень авторизации пользователя и разрешать запись или чтение только при наличии нужного флага.

## **Bilet de examinare Nr. 10**

### **1. Что понимаем под состоянием user и supervisor?**

Режим супервизора — привилегированный режим работы процессора, как правило используемый для выполнения ядра операционной системы.

В данном режиме работы процессора доступны привилегированные операции, как то операции ввода-вывода к периферийным устройствам, изменение параметров защиты памяти, настроек виртуальной памяти, системных параметров и прочих параметров конфигурации.

Как правило, в режиме супервизора или вообще не действуют ограничения защиты памяти или же они могут быть произвольным образом изменены, поэтому код, работающий в данном режиме, как правило, имеет полный доступ ко всем системным ресурсам (адресное пространство, регистры конфигурации процессора и т. п.). Во многих типах процессоров это наиболее привилегированный режим из всех доступных режимов.

Режим юзера - пользовательский режим (непривилегированный режим работы процессора, в котором исполняется код прикладных программ, и предоставляющий исполняемому в нем процессу доступ к системным ресурсам только посредством вызова системных сервисов)

### **2. В чем суть применения понятия расширенной машины в случае ОС?**

Использование большинства компьютеров на уровне машинного языка затруднительно. С этой точки зрения функцией ОС является предоставление пользователю некоторой расширенной или виртуальной машины, которую легче программировать и с которой легче работать, чем непосредственно с аппаратурой, составляющей реальную машину. Большинство современных компьютеров состоит из двух и более уровней.

Самый нижний уровень (нулевой) - аппаратное обеспечение машины.

Следующий уровень (первый) – микроархитектурный уровень. На этом уровне находятся регистры, которые формируют локальную память и АЛУ.

Следующий (второй) уровень составляет уровень архитектуры системы команд. Команды используют регистры и другие возможности аппаратуры. Команды формируют уровень ISA (Instruction Set Architecture), называемый машинным языком.

Следующий (третий) уровень обычно – гибридный. Большинство команд в его языке есть также и на уровне архитектуры системы команд. У этого уровня есть некоторые дополнительные особенности: набор новых команд, другая организация памяти, способность выполнять две и более программы одновременно и некоторые другие.

Четвертый уровень представляет собой символическую форму одного из языков низкого уровня (обычно ассемблер). На этом уровне можно писать программы в приемлемой для человека форме. Эти программы сначала транслируются на язык уровня 1, 2 или 3, а затем интерпретируются соответствующей виртуальной или фактически существующей (физической) машиной.

Уровни с пятого и выше предназначены для прикладных программистов, решающих конкретные задачи на языках высокого уровня.

### **3. Изобразите графически конфигурацию ПК (в рум вопросе не просят рисовать)**

Под конфигурацией понимают определенный набор комплектующих, исходя из их предназначения, номера и основных характеристик.

Зачастую конфигурация означает выбор аппаратного и программного обеспечения, прошивок и сопроводительной документации. Конфигурация влияет на функционирование и производительность компьютера. Существует 4 основные части оборудования персонального компьютера:

1. Материнская плата выполняет функцию координатора. Чипсет — микропроцессорный комплекс управляющий внутренними системами компьютера. Чипсет определяет основные возможности материнской платы. Набор проводников, по которым происходит обмен сигналами между внутренними устройствами компьютера. 2. Центральный процессор

3. Оперативная память (ОЗУ) отвечает за временное хранение данных при включённом компьютере. ПЗУ (постоянно запоминающее устройство) предназначен для длительного хранения данных при выключенном компьютере. 4. Блок питания

#### **Дополнительные:**

1. Жёсткий диск — основное устройство для долговременного хранения больших объемов данных и программ, выполняет специальное аппаратно-логическое устройство. К основным параметрам относятся ёмкость и производительность. 2. Звуковая плата

#### 4. Обязательные х-ки ОС с разделением времени

Доступность, надежность, безопасность, оптимальное использование ресурсов, качество интерфейса, простота адаптации, расширяемость.

#### 5. Какие операции выполняются при возврате из процедуры?

При возврате из процедуры выполняются следующие операции:

- На регистр указателя команды записывается адрес команды, с которой необходимо продолжить выполнение прерванной процедуры.
- Регистры состояния процессора восстанавливаются данными, находящимися в регистрах CR
- Регистры группы модуля компиляции считываются из таблицы модулей компиляции.

#### 6. Что представляет собой вызов режима супервизора?

Режим супервизора — привилегированный режим работы процессора, как правило используемый для выполнения ядра операционной системы.

В данном режиме работы процессора доступны привилегированные операции, как то операции ввода-вывода к периферийным устройствам, изменение параметров защиты памяти, настроек виртуальной памяти, системных параметров и прочих параметров конфигурации.

Как правило, в режиме супервизора или вообще не действуют ограничения защиты памяти или же они могут быть произвольным образом изменены, поэтому код, работающий в данном режиме, как правило, имеет полный доступ ко всем системным ресурсам (адресное пространство, регистры конфигурации процессора и т. п.). Во многих типах процессоров это наиболее привилегированный режим из всех доступных режимов.

#### 7. Опишите программу драйвера для синхронного ВВ

Драйвер - это программа, контролирующая базовую функциональность периферийного устройства.

Как правило, пользователь не взаимодействует с драйверами напрямую и вызывает функции ввода/вывода через сервисы высокого уровня. В этом случае между обработкой и передачей информации нет никакого параллелизма, и процессор остается занятым на протяжении всего периода передачи данных.

Предположим, что нам нужно вывести последовательность символов и массива с заданной длиной. Каждый последующий

символ может быть извлечен только если до него был извлечен стоящий перед ним символ. Для этого статус передачи проверяется в цикле на завершенность.

Синхронный ввод/вывод используется для простейших микропроцессоров или в тех случаях, когда процессор не может использоваться другим процессом в момент передачи данных.

#### 8. Опишите примитивы *fork* и *join*

Первыми примитивами, предложенными для динамического управления процессами, исторически были *fork* и *join*. Их использовали для организации параллельного исполнения программ в многопроцессорной системе. Для процедуры P инструкция *fork(p)*, исполняемая процессом p, создает новый процесс q, который выполняется параллельно с p.

Примитив *fork* представляет собой результат - идентификатор q (или null, если создание q невозможно). Первоначальный контекст q - копия p - счётчик, установленный на первую инструкцию p.

После создания процесса q инструкцией *fork* примитив *join(q)* позволяет родительскому процессу установить точку "рандеву" с q.

Исполнение *join(q)* блокирует родительский процесс до того момента, пока q не выполнит выход.

#### 9. Опишите модель производитель-потребитель и где он применяется.

Часто используемой схемой коммуникации является та, при которой некий процесс ("производитель") посылает сообщения другому процессу ("потребителю"), используя для этого буфер в общей памяти. Сообщения при этом имеют фиксированный размер, а буфер вмещает в себя N сообщений.

Требуемая синхронизация между производителем и потребителем обеспечивается ядром, обрабатывающим команды *write* производителя и *read* покупателя. Если производитель опережает потребителя (канал переполняется), ядро приостанавливает производителя при вызове *write*, пока в канале не появится место. Если потребитель опережает производителя (канал опустошается), ядро приостанавливает потребителя при вызове *read*, пока в канале не появятся данные.

## **Bilet de examinare Nr. 11**

### **1. Назначени технических устройств защиты памяти**

Для защиты памяти обычно используются следующие средства и механизмы:

- регистры границ памяти, устанавливающие нижний и верхний адреса оперативной памяти для программы, выполняемой в данный момент времени;
- "замки" защиты блоков памяти фиксированного размера в оперативной памяти; выполняемая программа заносит свой "ключ" в специальный регистр; каждая выборка и запись в оперативную память контролируется аппаратными средствами на подтверждение того, что ключ соответствует замку;
- сегментация памяти, представляющая использование дескрипторов для описания единиц данных в оперативной памяти; каждый дескриптор содержит начальный адрес сегмента, его длину и указатели, определяющие вид доступа к данным этого сегмента;
- страничная организация памяти, в которой каждой программе пользователя ставится в соответствие таблица страниц, отображающая виртуальные адреса в физические; обычно защита страничной организации памяти реализуется через сегментацию;
- иерархические кольца безопасности, которые обеспечивают аппаратную изоляцию данных и программ, относящихся к различным кольцам.

### **2. Опишите два метода использования понятия расширенной машины в случае ОС**

Первые операционные системы представляли собой одну большую программу. С эволюцией и усложнением ОС ввели концепцию расширенной машины. В случае ОС расширенную машину можно рассматривать в двух уровнях:

1. Внутренняя расширенная машина. Содержит ключевые функции ОС.
2. Внешняя расширенная машина. Содержит остальные функции ОС.

### **3. Назовите основные активности возникающие при разработке программы для ПК**

разработка программы;

- введение программы с использованием клавиатуры и текстового редактора;
- выполнение программы путем ввода необходимых данных с клавиатуры и извлечения результатов на дисплей или принтер;
- Изменение программы, если результаты не являются удовлетворительными и повторяют выполнение;
- завершение окончательной версии программы, включая документацию по ее необходимости;
- использование программы;
- запрос на запуск существующей программы.
- Просмотр результатов на экране, печать на принтер или их копирование в файл для последующего использования.

### **4. Состояние и контекст процессора**

Контекст процесса включает в себя содержимое адресного пространства задачи, выделенного процессу, а также содержимое

относящихся к процессу аппаратных регистров и структур данных ядра. С формальной точки зрения, контекст процесса объединяет в себе пользовательский контекст, регистровый контекст и системный контекст.

Пользовательский контекст состоит из команд и данных процесса, стека задачи и содержимого совместно используемого пространства памяти в виртуальных адресах процесса. Те части виртуального адресного пространства процесса, которые периодически отсутствуют в оперативной памяти вследствие выгрузки или замещения страниц, также включаются в пользовательский контекст.

Состояние процессора определяется содержанием регистров. Регистры бывают: адресными, регистрами данных, сегментными, системными, регистр состояния.

Регистр состояния содержит информацию касательно состояния процессора, режима функционирования, прерываний, касательно проведения текущего действия.

### **5. Как можно эмулировать отсутствующие операции?**

Некоторые инструкции в наборе команд процессорного ядра могут быть дополнительными и поэтому в

некоторых конфигурациях компьютера

могут отсутствовать. Попытка выполнения дополнительной отсутствующей операции генерирует драйвер типа недостающей инструкции.

Механизм прерываний может быть использован для реализации отсутствующих в базовом наборе операций.

#### **6. Опишите метод асинхронного ВВ**

Поток, выполняющий асинхронный ввод - вывод, отправляет запрос на ввод-вывод данных ядру.

Если запрос принят ядром, поток продолжает обрабатывать другое задание до тех пор, пока ядро не подаст сигналы потоку,

что операция ввода/вывода (I/O) полностью завершилась. Тогда поток прерывает работу со своим текущим заданием и обрабатывает данные от операции ввода/вывода (I/O) по мере необходимости.

#### **7. Что представляет собой точка синхронизации?**

Синхронизация процессов — приведение двух или нескольких процессов к такому их протеканию, когда определённые стадии разных процессов совершаются в определённом порядке, либо одновременно. Точками синхронизации называются такие точки на временной траектории процесса, в которых происходит синхронизация процессов.

Например, если мы поставим условие, что процесс Р не может начаться только после того, как завершится процесс Q, то точка в

которой это условие выполнится и будет точкой синхронизации.

#### **8. Опишите схему модели клиент-сервер**

Процесс сервера ассоциируется с воротами, куда клиенты предоставляют свои заявки.

Запрос клиента может требовать ответного результата. В таком случае клиент в своем запросе должен указать номер ворот,

в которых он будет заблокирован в ожидании результата.

## **Bilet de examinare Nr. 12**

### **1. Назначение технических средств прерывания**

Асинхронные, или внешние (аппаратные) прерывания — события, которые исходят от внешних источников (например, периферийных устройств) и могут произойти в любой произвольный момент: сигнал от таймера, сетевой карты или дискового накопителя, нажатие клавиш клавиатуры, движение мыши.

Факт возникновения в системе такого прерывания трактуется как запрос на прерывание. Аппаратные прерывания обеспечивают реакцию процессора на события, происходящие асинхронно по отношению к исполняемому программному коду.

Существует 2 типа аппаратных прерываний:

1. немаскируемые - обрабатываются всегда, независимо от запретов на другие прерывания. Используются для сигнализации о фатальных аппаратных ошибках.
2. маскируемые - прерывания, которые можно запрещать установкой соответствующих битов в регистре маскирования прерываний

Аппаратные прерывания используются для организации взаимодействия с внешними устройствами.

### **2. Что представляют собой уровни расширенной машины с точки зрения ОС?**

Использование большинства компьютеров на уровне машинного языка затруднительно. С этой точки зрения функцией ОС является предоставление пользователю некоторой расширенной или виртуальной машины, которую легче программировать и с которой легче работать, чем непосредственно с аппаратурой, составляющей реальную машину. Большинство современных компьютеров состоит из двух и более уровней.

Самый нижний уровень (нулевой) - аппаратное обеспечение машины.

Следующий уровень (первый) – микроархитектурный уровень. На этом уровне находятся регистры, которые формируют локальную память и АЛУ.

Следующий (второй) уровень составляет уровень архитектуры системы команд. Команды используют регистры и другие возможности аппаратуры. Команды формируют уровень ISA (Instruction Set Architecture), называемый машинным языком.

Следующий (третий) уровень обычно – гибридный. Большинство команд в его языке есть также и на уровне архитектуры системы команд. У этого уровня есть некоторые дополнительные особенности: набор новых команд, другая организация памяти, способность выполнять две и более программы одновременно и некоторые другие.

Четвертый уровень представляет собой символическую форму одного из языков низкого уровня (обычно ассемблер). На этом уровне можно писать программы в приемлемой для человека форме. Эти программы сначала транслируются на язык уровня 1, 2 или 3, а затем интерпретируются соответствующей виртуальной или фактически существующей (физической) машиной.

Уровни с пятого и выше предназначены для прикладных программистов, решающих конкретные задачи на языках высокого уровня.

### **3. Приведите типичную последовательность активностей, генерируемых при выполнении программы на ПК**

разработка программы;

- введение программы с использованием клавиатуры и текстового редактора;
- выполнение программы путем ввода необходимых данных с клавиатуры и извлечения результатов на дисплей или принтер;
- Изменение программы, если результаты не являются удовлетворительными и повторяют выполнение;
- завершение окончательной версии программы, включая документацию по ее необходимости;
- использование программы;
- запрос на запуск существующей программы.
- Просмотр результатов на экране, печать на принтер или их копирование в файл для последующего использования.

### **4. В чем суть метода пакетной обработки заданий?**

Пакетная обработка заданий - способ запуска задач на исполнение для избегания простоев

вычислительных систем. Фактически представляет собой список запускаемых программ с указанием параметров запуска и входных данных. Задачи запускаются последовательно.

## **5. Что называем асинхронными операциями?**

Асинхронные операции - такие операции, которые возвращают управление потоку сразу, не дожидаясь конца операции.

## **6. Как измеряется автоматически объем памяти ПК?**

Память компьютера - совокупность устройств для хранения информации.

Различают два вида памяти - внутреннюю и внешнюю (по отношению к материнской (системной) плате). Внутренняя память компьютера является быстродействующей, но имеет ограниченный объем.

Работа с внешней памятью требует гораздо большего времени, но она позволяет хранить практически неограниченное количество информации.

Внутренняя память состоит из нескольких частей: оперативной (ОЗУ), постоянной (ПЗУ) и кэш-памяти. Основные характеристики памяти: 1. Объем (емкость) 2. Быстродействие (время доступа)

Память компьютера построена из двоичных запоминающих элементов — битов, объединенных в группы по 8 битов, которые называются

байтами. Байты могут объединяться в ячейки, которые называются также словами. Широко используются и более крупные производные

единицы объема памяти: Килобайт, Мегабайт, Гигабайт, а также, в последнее время, Терабайт и Петабайт.

## **7. Специфицируйте задачу буферизированного ВВ**

Традиционным способом снижения накладных расходов при выполнении обменов с устройствами внешней памяти, имеющими блочную структуру, является буферизация блочного ввода/вывода. Это означает, что любой блок устройства внешней памяти считывается прежде всего в некоторый буфер области основной памяти, называемой в ОС UNIX системным кэшем, и уже оттуда полностью или частично (в зависимости от вида обмена) копируется в соответствующее пользовательское пространство.

Существуют три причины, приводящие к использованию буферов в базовой подсистеме ввода-вывода:

-Первая причина буферизации — это разные скорости приема и передачи информации, которыми обладают участники обмена.

-Вторая причина буферизации — это разные объемы данных, которые могут быть приняты или получены участниками обмена одновременно.

-Третья причина буферизации связана с необходимостью копирования информации из приложений, осуществляющих ввод-вывод, в буфера ядра операционной системы и обратно.

## **8. Мотивируйте необходимость введения состояния wait и опишите возникшие**

**при этом переходы**

Введение состояния wait необходимо при синхронизации процессов.

Синхронизация процессов — приведение двух или нескольких процессов к такому их протеканию, когда определённые стадии разных процессов совершаются в определённом порядке, либо одновременно.

В случае, если несколько процессов используют одни и те же переменные, мы не можем запустить один из них до тех пор, пока

другой не завершится, для избежания ошибок. В таком случае используется wait.

Так же возникли такие состояния как: ready и exe. Exe означает, что процесс выполняется в данный момент. Ready, что процесс может

быть запущен, а wait, что процесс не может быть запущен.

## **9. В чем суть динамического управления процессами?**

Только в самых простых системах процессы являются числом постоянным и создаются одновременно с инициализацией системы. Динамическое управление процессами подразумевает, что процесс может создаваться в различных точках работы системы. Создание процесса предполагает распределение ресурсов и инициализацию контекста. А уничтожение процесса освобождает все используемые ресурсы.