**1.Кои техники придонесуваат за подобрување на перформансите на компјутерот?**

**2.Што е кеш меморија и кои видови на локалност на мемориски адреси се ползуваат?**

-Кеш меморија претставува мала брза меморија во која се чуваат најскоро употребуваните мемориски зборови, со што се забрзува пристапот до нив . Кеш меморијата е поделена на блокови со фиксна големина – кеш линии (cache lines), и при појава на промашување, цела кеш линија се вчитува од главната меморија.

-Постојат:

А) Просторна локалност – на мемориските локации чии адреси се нумерички слични на мемориската локација кон која се пристапило неоддамна со голема веројатност повторно да им се пристапи во блиска иднина.

Б) Временска локалност – на мемориските локации кон кои се пристапило неоддамна со голема веројатност повторно ќе им се пристапи во блиска иднина.

**3.Што е DMA контролер?**

-Чип со директен пристап до магистралата која има најмалку 4 регистри чија содржина може софтверски да се менува:

1. Мемориска адреса од која чита или запишува
2. Број на бајтови или зборови што треба да се пренесат
3. Идентификатор на влезо / излезниот уред
4. Вид на операцијата читање – 0 или запишување 1

**4.Што е контролна зависност, а што вистинска податочна зависност. Како може да се минимизира ефектот од нивното појавување?**

-Контролна зависност – извршувањето на низа инструкции зависи од исходот на инструкцијата за разгранување. Ефектот од контролната зависност може да се минимизира со: шпекулативно извршување на контролно зависните инструкции засновано на претскажување на разранувањето.

-Вистинската податочна зависност – некоја инструкција консумира вредност генерира од претходната инструкција во низата.

-Шпекулативно извршување на податочно зависните инструкции засновано на претскажување на излезните вредности на инструкциите.

**5.Кои начини на адресирање постојат?**

1. Непосредно: Автоматски се превзема меморија со превземање на самата инструкција. Наместо адреса инструкцијата веќе го содржи операндот (операндот не треба да се бара ниту во меморија, ниту во регистер).
2. Директно:Во инструкцијата ни се содржи адресата на мемориската локација на која што се наоѓа операторот. Операндот се наоѓа во меморија, а мемориската адреса е директно наведена
3. Регистарско: Исто како кај директното, само се специфира регистер наместо емориска локација, операндот се наоѓа во регистер, а регистерот е директно наведен
4. Регистарско – Индиректно: Адресата на операндот се содржи во некој регистер. Тука адресата се користи како покажувач, операндот се наоѓа во меморија, но мемориската адреса треба да се прочита од некој регистер (регистерот е покажувач)
5. Индексирано: Мемориските локации се адресираат со наведување на регистер и константно релативно поместување offset.Операндот се наоѓа во меморија, но адресата треба да се добие со собирање на две вредности (регистер + константа)
6. Базно – Индексирано: Мемориските адреси се пресметуваат со собирање на два регистри(база и индекс) и релативното поместување, операндот се наоѓа во меморијата, но адресата треба да се добие со собирање на две или три вредности (регистер + регистер + константа)
7. Stack – адресирање: нула адресни инструкции

**6.Што е стапица (trap), а што прекин (interrupt) и кои се разликите помеѓу нив?**

-Стапица е еден вид автоматско повикување на одредена процедура (trap handler) иницирано од некоја состојба предизвикана од програмата, а детектирана од хардверот или микропрограмата,а прекин е промена на текот на контролата предизвикана не од програмата која се извршува туку од други причини ( најчесто поврзано со влезно излезни операции).

-Разлика: стапиците се синхрони во програмта (предизвикани дирекно од неа) при исти влезни податоци секогаш се појавуваат на исто место во програмата, а пак прекините се асинхрни(во најдобар случај индиректно се повикани од програмата) нивното појавување варира и може да зависи, на пример од тоа кога точно операторот на терминалот ќе притисне Еnter.

**7.Во колку и кои категории можат да се поделат инструкциите од нивото на архитектура на иснтрукциското множество?**

-Постојат 6 типови на инструкциско множество:

1. Инструкции за копирање на податоци
2. Дијадни операции
3. Монадни операции
4. Споредби и условни разгранувања
5. Инструкции за повикување на процедури
6. Влезни-излезни инструкции

**8.Кои влезно-излезни шеми се користат кај персоналните компјутери?**

1. Програмиран влез/излез
2. Влез/излез реализиран со прекини (interrupts)
3. Влез/излез реализиран со директен пристап во меморија (DMA)

**9.Што е дадотека (file) и како се реализираат влезно/излезните операции?**

- Датотеката е апстракција, која во наједноставната форма, претставува низа од бајтови запишани на некои влезно излезни уреди. Влезно излезните опера се реализират со системски повикувачи за отварање,читање,запишување и затварање.

**10.Што е анти-зависност, а што излезна зависност и како можат да се елеминираат?**

1. Анти-зависност – инструкцијата се обидува да запише резултат во регистер чија содржина сеуште не е прочитана од претходната инструкција.
2. Излезна зависност – инструкцијата се обидува да запише резултат во регистер во кој запишува и некоја претходна инструкција.

**11.Објаснетего секој од следните поими!**

**- Преведувач** – секоја инструкција од програмата напишана во јазикот L1 се заменува со соодветна низа од инструкции од јазикот L0, потоа компјутерот ја извршува новата L0 програма наместо старата L1 програма

**- Интерпретатор** – е програма кога предходно е напишана во L0 јазикот и ги прегледува (една по една) инструкциите од програмата напишана во јазикот L1 и веднаш ги извршува соодветните низи од L0 инструкции

**- Виртуелна машина** – наместо да се размислува за преведување или интерпретирање, поедноставно е да се замисли дека постои хипотетички компјутер-виртуелна машина чиј машински јазик е јазикот L1. Тогаш човекот може да пишува програми за виртуелната машина, исто како таа да постои во стварноста.

**12.Кои принципи на дизајн се применуваат кај современите процесори?**

-Дирекно хардверско извршување на инструкциите, максимизирање на интензитетот на издавање (issue) на инструкциите, едноставно декодирање на инструкциите, обраќање до меморијата со посебни инструкции и обезбедување на голем број регистри.

**13.Направи разлика меѓу процесорско поле и вектор процесор?**

-Процесорското поле служи за реализација на операциите на собирање, постојат онолку единици за собирање колку што има парови од податоци, додека Векторскиот процесор служи за, сите операции на собирање на паровите од податоци ги изведува една единствена високо-проточна единица за собирање.

**14.На колку начини можат да бидат нумерирани бајтовите во еден збор?**

На два начини: од лево на десно (big endian) компјутер и од десно на лево (little endian) компјутер.

**15.За што служи секундарната меморија и кои се единици на секундарната меморија?**

-Служи за чување на големи количества податоци во подолг временски период. Капацитетот е од прилика неколку терабајти. Единици на секундарната меморија: магнетна лента, магнетен диск, дискета, компакт диск, дискета, CD-ROM, DVD, FMD-ROM, дигитален повеќенаменски диск, Blu-ray, флуросцентен повеќеслоен диск и други напредни технологии.

**16.Како се запишуваат/читаат податоците на/од оптички диск?**

-Запишувањето се врши со помош на сторго насочен ласерски зрак, со кој се прават микроскопски вдлабнатини на површината на рефлективна пластична плоча. Читањето се врши со друг ласер со дгруг ласер со кој се осветлува површината на дискот.

**17.Која е намената на следните регистри кај Mic-1 архитектурата: MAR, MBR, SP, CCP, OPC и MPC?**

-MAR (Memory Address Register) – мемориски адресен регистер кој содржи адреси на мемориски зборови

-MBR – мемориски бајт регистер

-SP (Stack Pointer) – покажува на врвот од stack-от за тековната процедура

-CCP (Constant Pool register) – ја содржи адресата на првиот збор од базенот/подрачјето со константи

-OPC – привремен регистер со недефинирана намена

-MPC (Micro Program Counter) – микропрограмски бројач (адресен регистар за контролната меморија)

**18.Во форматот на Mic-1 микро-инструкциите, регистерот од магистрала B е кодиран со 4-битно поле(зошто)?**

-Никогаш не е пожелно проследување на содржината на повеќе од еден регистер од магистралата В, 9-те контролни сигнали кои одредуваат од кои регистри може да се чита можат да се кодираат со 4-битен сигнал ( тој по декодирањето има 16-бита но ние користиме само 9).

**19.Во колку и кои фази се одвива протечната обработка кај Mic-4 архитектурата?**

-Се одвива во 7 фази:

1. Единицата за преземање на инструкции IFU
2. Единицата за дeкодирање Decoder
3. Единицата со ред на чекање Queue
4. Лоцирање и преземање на операнди Operands
5. Извршување на инструкции Exec
6. Запишување во регистри Write
7. Запишување во меморијата Memory

**20.Кои техники придонесуваат за подобрување на перформансите на процесорот?**

-Тоа се кеш меморија (cashe), извршување out-of-order (вон редослед) со преименување на регистрите, спекулативно извршување кое може да претскажува вредности и разгранувања.

**21.Која е разликата меѓу преведување и интерпретација?**

-Кај преведувањето секоја инстукција од програмата напишана во јазикот L1 се заменува со соодветната низа од инструкции од јазикот L0 и потоа компјутерот ја извршува новата L0 програма немсто старата L1 програма, додека Интерпретација претставува прегледување на инструкциите една по една од програмата напишана во јазикот L1 и веднаш ги извршува соодветните низи од L0 инструкциите.

**22.Од колку и кои нивоа се состојат повеќето современи компјутери?**

-Се состојат од 2 или повеќе, падури и до 6 нивоа:

1. Дигитална логика
2. Микроархитектура
3. Архитектура на инструкциски множество
4. Оперативен систем
5. Асемблерски јазик
6. Проблемски-ориентирани јазици

**23.Кои се карактеристики на суперскаларните архитектури?**

-Може да се превзема неколку инструкции одеднаш и секоја од нив одеднаш да ја упатува кон посебни единици за декодирање и извршување, инструкциите можат да се извршуваат паралелно доколку инструкциите не се во конфликт околу употреба на ресурси и никоја инструкција не смее да зависи од резултатот на друга инструкција.  
  
- Една единица за преземање на инструкции (IFU) презема голем број инструкции и ги упатува во различни единици за декодирање, така што се извршуваат паралелно во еден циклус

- Современите суперскаларни се со повеќе извршни единици наместо IFU, бидејќи се претпоставува дека на извршните единици им се даваат инструкции побрзо отколку што можат да ги извршат.

**24.Што е мултипроцесор, а што мултикомпјутер?**

-Мултипроцесор е систем од повеќе процесори (секој со своја контролна единица) со заедничка меморија, а Мултикомпјутер е систем од голем број меѓусебно поврзани компјутери, секој со своја сопствена меморија, не постои заедничка меморија со цел да се избегне потребата од поврзување на сите процесори.

**25.Кои се општи карактеристики на секундарната меморија?**

Содржината е постојана, потребно е многу подолго време за пристап до податоците, многу поисплатлива, може да биде сместена на различни медиуми со примена на различни технологии, има поголема застапеност на методи за директен пристап, поголем капацитет, помали трошоци и зголемена преносливост.

**26.Како се запишуваат/читаат податоците на/од магнетен диск (хард диск)?**

-Хард дисковите читаат од и запишуваат на една или повеќе вртливи магнетно обложени алуминиумски плочи поставени на заедничка оска и заштитени од надворешни и атмосферски влијанија (херметички затворени). За секоја плоча постојат по две глави за читање/запишување (по една на секоја страна) кои лебдат над површината на растојание помало од 25 микрони и се придвижуваат од периферијата кон центарот и обратно. На секоја од плочите податоците се запишуваат во коцетрични патеки, секоја патека е поделена на сектори. До секој податок може да се пристапи директно ако се знае бројот на патеката , секторот и главата.

**27.Која е намената на следните регистри кај Mic-1 архитектурата: MDR, LV, PC, TOS, H и MIR?**

-MDR (Memory Data Register) – мемориски податочен регистер

-LV (Local Variable Pointer) – покажува на дното од stack-от за тековната процедура

-PC (Program Counter) – програмски бројач кој вчитува 1 бајт од меморијата и го сместува во 8-те најмалку значајни битови во регистерот MBR, содржи адреси од бајтови

-TOS - регистер кој ја содржи вредноста од она мемориска локација кон која покажува SP, т.е зборот што е на врвот на стакот.

-H – го содржи едниот податок врз кој треба да се изврши операцијата заедно со податокот кој патува од A магистралата

-MIR (Micro Instruction Register) – микроинструкциски регистер за сместување на тековната микроинструкција

**28.Кои подрачја се присутни во меморијата на IJVM?**

-Присутни се: подрачја на константи, го сочинуваат константи,стрингови и покажувачи кон други мемориски подрачја, се полни кога програмата вчитува во меморијата и потоа не се менува -регистер СРР,   
рамка на локални променливи – регистер LV,   
stack за операнди – регистер SP,   
подрачје на методи – ја содржи програмата, има тертман на поле од бајтови – регистер PC.

**30.Објасни ги така наречените RISC принципи за дизајн кои се применуваат кај современите компјутери!**

- RISC (Reduced Instruction Set Computer) претставува компјутер со редуцирано инструкциско множество. Тој врши директно хардверско извршување на инструкциите, не се интерпретираат со микроконструкции, но наспроти нив кај CISC (Complex Instruction Set Computer) покомплексните инструкции треба да се разбиваат на делови, кои потоа ќе бидат извршувани како низи од микроинструкции.

**31.Кои техники и со каква конкретна цел се применуваат за да се зголеми брзината на извршување на инструкциите (Mic-1 -> Mic-2 -> Mic-3)?**- Микроархитектура со три магистрали (А,B,C)  
- Посебна единица за преземање на инструкции (IFU)  
- Протечна обработка (pipeline)

**32.Објасни ги разликите помеѓу непосредно, директно и регистарско адресирање!**- Непосредно: Автоматски се превзема меморија со превземање на самата инструкција. Наместо адреса инструкцијата веќе го содржи операндот (операндот не треба да се бара ниту во меморија, ниту во регистер).

- Директно:Во инструкцијата ни се содржи адресата на мемориската локација на која што се наоѓа операторот. Операндот се наоѓа во меморија, а мемориската адреса е директно наведена

Регистарско: Исто како кај директното, само се специфира регистер наместо емориска локација, операндот се наоѓа во регистер, а регистерот е директно наведен

**33.Објасни како се реализира влез/излез со директен пристап до меморијата!**- DMA e чип со директен пристап кон магистралата и има најмалку четири регистри чија содржина може да се менуваат софтверски. По запишувањето на секој бајт, DMA го инкрементира сопствениот за 1 и го декрементира бројачот на бајтови за 1. Ако бројачот сеуште е поголем од 0, се чита следниот бајт од меморијата и се пренесува до терминалот. Кога бројачот е 0, трансферот на податоци завршува и се генерира прекин.   
Ги има следните регистри  
а) Идентификатор за IO уред  
б) знак за дали ќе се чита(0) или запишува (1)  
в) адреса од каде се чита/пишува  
г) број на бајтови кои се пренесуваат

**34.Кои се придобивките од внесувањето на три дополнителни регистри (latches) вдолж секоја од магистралите на податочната патека ( на пр. Mic-3)?**

**35.Објасни како се реализира програмиран влез/излез!**

- Обично постои една влезна и една излезна инструкција. Секоја од нив може да селектира еден од влезно/излезните уреди. Еден единствен знак се пренесува помеѓу одреден регистер во процесорот и селектираниот влезно/излезен уред. Процесорот мора да изврши секвенца од инструкции за секој знак што се чита или запишува.

**36.Што претставува процесорско поле и што означува кратенката SIMD?  
-** Процесорското поле - за реализација на операциите на собирање, постојат онолку единици за собирање колку што има парови од податоци

**37.Третата фаза од протечната обработка кај Mic-4 микроархитектурата содржи единица со ред на чекање (queuing unit). Опиши ја нејзината работа!  
Единицата со ред на чекање треба да се најди во презентаците**

**38. Што е виртуелна машина и во колку нивоа (layers, levels) можат да се набљудуваат повеќето современи компјутери?**

- **Виртуелна машина** – наместо да се размислува за преведување или интерпретирање, поедноставно е да се замисли дека постои хипотетички компјутер-виртуелна машина чиј машински јазик е јазикот L1. Тогаш човекот може да пишува програми за виртуелната машина, исто како таа да постои во стварноста.

- Бидејќи секој од јазиците се потпира на неговиот претходник, компјутерот може да се набљудува како серија од нивоа (layers, levels), поставени едно врз друго, од наједноставното до најсофистицираното и така до безкрај.

**39.Каква промена на текот на контролата (control flow) предизвикуваат процедурите, а каква корутините?**

**40.   
- Директно адресирање** :Во инструкцијата ни се содржи адресата на мемориската локација на која што се наоѓа операторот. Операндот се наоѓа во меморија, а мемориската адреса е директно наведена

**- Регистарско адресирање**: Исто како кај директното, само се специфира регистер наместо емориска локација, операндот се наоѓа во регистер, а регистерот е директно наведен

**- Базно** – Индексирано адресирање: Мемориските адреси се пресметуваат со собирање на два регистри(база и индекс) и релативното поместување, операндот се наоѓа во меморијата, но адресата треба да се добие со собирање на две или три вредности (регистер + регистер + константа)

**41.  
- Просторна локалност кај кеш меморијата значи -** на мемориските локации чии адреси се нумерички слични на мемориската локација кон која се пристапило неоддамна со голема веројатност повторно да им се пристапи во блиска иднина.

**- Временска локалност кај кеш меморијата значи –** на мемориските локации кон кои се пристапило неоддамна со голема веројатност повторно ќе им се пристапи во блиска иднина.

**43.Непосредно адресирање -** Автоматски се превзема меморија со превземање на самата инструкција. Наместо адреса инструкцијата веќе го содржи операндот (операндот не треба да се бара ниту во меморија, ниту во регистер).

**- Регистерско-индиректно адресирање -** Адресата на операндот се содржи во некој регистер.Тука адресата се користи како покажувач, операндот се наоѓа во меморија, но мемориската адреса треба да се прочита од некој регистер (регистерот е покажувач).

**- Индексирано адресирање -** Мемориските локации се адресираат со наведување на регистер и константно релативно поместување offset.Операндот се наоѓа во меморија, но адресата треба да се добие со собирање на две вредности (регистер + константа).

1. **Кои начини на адресирање (адресни режими) постојат?  
   -** Непосредо  
   - Директно  
   - Регистарско  
   - Регистарко - Индиректно  
   - Индексирано  
   - Базно - Индексирано  
   - Stack - адресирање

**45.Кои механизми се применуваат за синхронизација на нишки (threads) кај оперативниот систем Windows NT?**- Се применуваат: Семафори, MUTEX, Критични секции, Настани

**46.Објасни ги поимите:**

**- Процесорско поле** – за реализација на операциите на собирање, постојат оноклу единици за собирање колку што има парови од податоци

**- Векторски процесор** – сите операции на собирање на паровите од податоци ги изведува една единствена високо-проточна единица за собирање

**- Мултипроцесор** – систем од повеќе процесори (секој со своја контролна единица) со заедничка меморија(shared memory)

**- Мултикомпјутер** – систем од голем број меѓусебно поврзани компјутери, секој со своја сопствена меморија, не постои заедничка меморија со цел да се избегне потребата од поврзување на сите процесори

**47.Што е n-насочна ограничено асоцијативна кеш меморија и кој алгоритам се применува за определување на кеш линијата што треба да се исфрли при преземање на нова кеш линија од главната меморија?**- Тое е кеш меморија со n-можни позиции за секоја адреса, а кога треба да се исфрли некоја линија за да се донесе нова се користи LRU (Least Recently Used) алгоритамот односно линија која најдолго време не била употребувана.

**48.Кои принципи на дизајн се применуват кај современите компјутери?  
- Директно хардверско извршување на инструкциите**

**- Максимизирање на интензитетот на издавање на инструкции (број на инструкции кои можат да започнат со извршување во една секунда)**

**- Едноставно декодирање на инструкциите**

**- Обраќање до меморијата со посебни инструкции**

**- Обезбедување на голем број регистри  
Два пати го има истиот одговор, да се провери дали одговорот е за кај современите компјутери или процесори од презентаците**

**49.Хиерархиски подреди ги видовите на меморија и објасни кои клучни параметри се менуваат и како! Да се провери  
-** Tape и оптички диск, магнетен диск, главна меморија, кеш и регистер.  
 **-** Помало време/растојание на пристап (Поголема брзина)  
**-** Поголем капацитет (помала цена по единица меморија)

**50. Од што се состои основната операција на податочната патека и како е контролирано нејзиното функционирање?**

- Податочната патека ( data path ) е онај дел од процесорот кој ги содржи аритметичко-логичката единица (ALU), нејзините влезови и нејзините излези. Содржи одреден број регистри меѓу кои може да ги споменеме:  
  
- MAR - мемориски адресен регистер  
- MDR - мемориски податочен регистер  
- PC - програмски бројач  
- MBR - мемориски бајт регистер  
- SP - Стек поинтер  
- LV - Local Variable Pointer   
- CPP - Constant Pool Pointer  
- TOS - врв на магацинот  
- OPC - привремен регистер со недифинирана надмена  
- H - холдинг регистер кој е лев влез во ALU  
  
Нејзиното функционирање: содржината на овие регистри може да биде прочитана преку магистралата B која е десен влез во аритметичко-логичката единица, а додека пак нејзиниот излез е поврзан со поместувач кој со магистралата C е поврзан со регистрите. Функцијата што треба да ја изврши ALU се определува со 6 контролни линии.

**51.Врз основа на која карактеристика на инструкциите се врши претскажување на вредностите и кои шеми се применуваат?  
-** Претскажување засновано на контекс хибридно претскажување и претскажување на последно забележана вредност. А шемите се адаптивен претскажувач со две нивоа, и хибриден претскажувач.

**52.Кои критериуми се применуваат при дизајнирањето на форматот на инструкции?**Се применуваат следните критериуми:   
- минимизирање на инстукциите (помал мемориски простор, поголем број инстукции во единица време)  
- Обезбедување доволно простор за специфицирање на сите можни операции во форматот на инстукцијата  
- Компромис помеѓу должината на основната мемориска единица и должината на адресите и инстукциите.

**53.Објасни ги термините:**

**- thrashing** - е појава при која една програма често и непрекинато предизвикува грешки поради не присуство на бараната страница

**- внатрешна фрагментација** - е кога програмата и податоците ги исполнуваат страниците речиси секогаш има неискористен простор така ако страниците се со големина n бајти n/2, неискористен мемориски простор, па ако се користат помали страни имаме предност што имаме помалку губиток на меморија и помал thrashing ефект, но неефикасно ја искористуваме пропустната моќ дискот и имаме голем број на страни и голема табела

**- надворешна фрагментација** - кај надворешната фрагментација се јавува неискористен простор помеѓу сегментите (празнини) и затоа се користат некои методи: се истуснуваат празнините односно сегментите се предвижуваат кон мемориската локација 0 при појавување на нова празнина, потоа best fit - се бара најмалата празнина во која може да го собере сегментот, first fir - сегментот се сместува во првата празнина која може да го собере.

**54. Опиши ја секвенцата од мали чекори (превзми-декодирај-изврши) – fetch-decode-execute при извршување на инструкциите.**

-Превземи ја инструкцијата од меморијата во инструкцискиот регистер.

-Промени го програмскиот бројач, така што ќе се покажува следната инструкција.

-Определи го типот на тукушто превземената инструкција.

-Ако инструкцијата користи збор од меморијата, определи каде се наоѓа зборот.

-Ако е потребно, превземи го зборот во еден од регистрите.

-Изврши ја инструкцијата

-Врати се на чекор 1 и започни со извршување на следната инструкција.

**55. Ако собирањето на две групи од по 10 броеви е претставено на векторско поле :**

**Repeat this loop 10 times**

**Read the next instruction and decode it**

**Fetch this number**

**Fetch that number**

**Add them**

**Put the result here**

**End Loop**

* Векторски процесор:
  + Read instruction and decode it
  + Fetch these 10 numbers
  + Fetch those 10 numbers
  + Add them
  + Put the results here

**56. Да се открие грешката во 21-битниот кодиран збор 001011100000101101111 (чекор по чекор)**

-При конструирањето на код за корекција на грешки, со примена на Хаминговиот алгоритам за корекција на 16-битен збор му се додаваат 5 битови за парност.

**57. Ако времето на пристап до кеш меморијата изнесува 4 нс и ако 3500 од вкупно 4000 микроинструкциски обраќања се реализираат во кеш меморијата, средното време на пристап до меморискиот подсистем изнесува 14нс. Да се определи времето на пристап до главната меморија.**

-

c-Време на пристап до cashe меморија

m-Време на пристап до главната меморија

h-Процент од сите мемориски обраќања кои можат да бидат реализирани во кеш меморијата (hit ratio – интензитет на погодување)

tsr- средно време на пристап

c=4ns

h=3500/4000

tsr=14ns

m=?

tsr=c+(1-h)\*m

14=4+(1-3500/4000)\*m

14=4+(1-0.875)\*m

14=4+0.125\*m

14-4=0.125\*m

10=0.125\*m

m=10/0.125

m=80

**58. Што содржи податочната патека кај Mic-1 микроархитектурата на IJVM и како се определува функцијата што треба да ја изврши аритметичко-логичката единица ?**

- Податочната патека ги содржи аритметичко-логичката единица (ALU), нејзините влезови и излези и содржи одреден број (32 битни ) регистри со соодветни имиња: PC, SP, MDR..

- Функцијата што треба да изврши аритметичко-логичката единица се определува со 6 контролни линии:

F0 и F1 – ја определуваат аритметичко-логичката операција

ENA и ENB - ги овозможуваат (enable) левиот и деснит влез на ALU

INVA – го инвертира левиот влез

INC – го инкрементира односно зголемува за 1 излезот

**59. Што е Garbage Collector ?**

- Тоа е парче софтвер кое бара објекти на heap-от кои повеќе не се во употреба, со цел да ослободи дополнителен мемориски простор ( при појава на overflow кај heap-от).

**60. Од колку нивоа се состојат повечето современи компјутери ?**

- Се состојат од 2 или повеќе, па дури и до 6 нивоа:

1.дигитална логика

2.микроархитектура

3.архитектура на инструкциско множество

4.оперативен систем

5.асемблерски јазик

6.проблемски-оринтирани јазици