

2024 09 23



유연한 사고방식

흐름 flow이해

이렇게이렇게 짜면 되는거 아니야? flow흐름 종이에 적고 각 함수의 역할을 종이에 적어보기



CPU(\instruction set archetect)=ISA -무조건 연산(사칙연산)밖에 못함. 이
동 복붙에 대한 명령어

cpu생산하는 회사 /설계하는 회사

)설계하는 회사가 설계도를 여러회사에 뿌림 그래서 ISA는 회사마다 다름(64비
트,32비트)

여기서 ★★비트는 명령어 크기, 명령어가 복잡하다는것은 길이가 길다는 뜻
(면접질문)

((아키텍처 구조 관련 질문 많이함.)) Q왜 안돌아가는것에 대해 설명을 할 수 있어
야함.

휴대폰cpu가 pc에 안돌아가는 이유 명령어셋이 달라서

CPU와 SSD만 있으면 동작잘함. ((근데 램도 있어야 함 ram은 랜덤 액세스 메모
리임 cpu가만히 있는 걸 방지 즉 속도를 보충하는 역할 더 빠르게 해줌.)

cache메모리 cash라는 다리도 있음 .ram은 용량 작음 (ram은 국자임 크면클수
록 좋음 저수지 물퍼오기) miss(1번저수지에 퍼와야 하는데 3번 저수지에 퍼옴)
hit(는 일치하느것)

cash메모리는 ram보다 빠름

SSD HDD(저수지) ram의 크기에 따라 달라짐 (약수터 의 물통크기)

main(mother board)-매우 중요함. 차의 프레임 (부품을 연결해주는데 단단해
야지)메인보드에따라 성능이 달라짐(각부품을 연결해줌 버스라고도 함) 데이터만
이동하는 데이터버스 명령어만 이용하는 데이터버스가 있음(소프트웨어적 개념)
하드웨어적 개념은 pcie(레인이라는 뜻, 데이터를 한꺼번에 짹보내야하니까 레인
의 수가 많아야함.)

powr은 부팅관리. Bios에서 firmware(시작순서,방법이 기록된것)
Bios(역할:데이터(부품들의 역할)

risc 명령어 셋이 작은거- 핸드폰, 인베리드 시스템 전력을 무한정 내면 됨(소비
전력 적음)

cisc 명령어 셋이 큰거(cpu 안에 스위치가 많다.)그만큼 발열이 많아서 안좋음.
소비전력도 많다

데이터 는 0과 1로 특정 번지에 저장이 됨(2진수로 바꿔줌 컴파일러 번역기)

트랜지스터들이 모여 한명령어를 만듦

모든 명령어는 스위치로 형성됨 (명령어는 스위치의 집합)

RAM MB SSD(HDD)-보조기억장치(용량이 딸립니다. SSD 를 교체하셔야 되요)

Solidstate Drive(반도체)

Hard disk drive-자성 1과 0기록

intel이라는 명령어 에이어램(스마트 폰 cpu명령어셋)——하나하나 구현을 다해야 함 딜레이 생겨서 속도가 느려짐 그래서 중국 자동차냥

power

GPU

작업장 ?게임에서 아이템 수집해서 돈벌

윈도우 켜기→메인보드 신호→ 파워(신호)→메인보드에 전력을 따다닥→ 채굴장 사람은 메인보드하나에다가 파워를 여러개 연결 (클리비온 연결해서) →펌웨어 바이오스가 어떤 부품이 있는걸 알기 때문에 펌웨어에서 순서나 부팅방법을 키면서 →부팅영역이 있는 기억장치 (HDD,SSD)efi가 부팅영역임

부팅영역이란?

하드디스크는 0과 1로 이루어짐

저장하는 방식이 다름 (mac,리눅스,윈도우,0,1을)

맥 윈도우 함께쓰는사람은 외장하드를 2개 들고 다녀야함.

efi -파일저장형식(0과1을 저장하는 형식)

SSD는 섹션

efi는 500메가 정도면 충분

드라이버 - 역할)하드웨어 제어하는 소프트웨어 역할

efi → kernelshell(조개껍데기)

파일탐색도 불러와지는 것

kernel은 드라이버들의 집합체, 실제드라이브들이 동작을 함.

kernel과 인터페이스 GUI에서 받은걸 kernel단위로 명령어를 내려줘야함.

cmd(윈도우) terminal(리눅스)

부팅순서를 올려줌 (efi를 올려야 됨)(내리면 안켜짐)

듀얼 부팅의 원리 윈도우-리눅스 이런거 ..커널도 2개 하드디스크 충돌일어나면 골치
efi영역이 2개도 될수 있네??

—>컴터 이론은 알아야 됨.

이

임베디드???

쿨러(굴러가는팬)청소해줘야함

같은 경우ssd없는 경우있음 ram에만 다해놔서



program 실행되고 만드는 과정

프로그래밍 언어로 만들어진 파일 + data 들의 집합

작성한다음 컴퓨터가 알아들을 수 있도록 번역 (

compile번역

compiler 번역기

파일 다운 exe 형식(파일확장자:어떤형식으로 구성되어있는지 알려줌) 확장자는 없어도됨?

알려주는 역할만해서 필요가 없다.

shell 스크립트

리눅스는 확장자 안써도 실행됨.

`.exe` 파일은 Windows 운영 체제에서 실행 가능한 프로그램 파일

exe (code, compiler, data, 요3가지만 있으면)

파일 확장자 이름이 bin(binary의 약자)

더블클릭하면 bin파일임.

프로그램 삭제하더라도 찌꺼기가 남아있는 경우가 많음

맥이 오래써도 빠름

x64 x32(미리 컴파일된 bin 파일임)설치할 필요없음.

한컴은 컴파일러 설치가 되어있어서 오류가 많이 나는 경우가 많음.

window는 설치가 잘 안됨 (설치할 수 있는 방법 은 >> 바이너리 파일 버전 마다 파일 하나씩 복사해서 하면 됨)

exe 파일도 코드로 되어있음

.exe

압축→컴파일 무슨 언어인가?? shell script

shell script

exe bat linex bash.sh

cmd

명령어가 다다름.

요즘 powershell이라는 걸 밀고 있음.

응용프로그램 복사하면 설치가능

shell에서 동작하는 것은 시스템 명령어이다 (compiler)
window컴파일러 만드는 것이 좋다.

kernel driver python .net ((cpu)) cpu 호환성
중간 파일을 만들어주면 언어를 만드는데 수고를 덜어준다.
링킹?

메모장에다가 침(
윈도우는 쉘컴파일러 자체가 시스템임

object파일은 링킹하는 과정에서 빠진거 있으면 라이브러리가



윈도우 리눅스 메모장 방식 다름

뭐가 다름?? 인코딩이 다름

엔터 인코딩이 달라서 우리가 치는 언어나 문자같은 것들을 16진법의 숫자로 바꿔주는 것을 의미

노트

인코딩이랑 컴파일이란 비슷한 개념

인코딩 약속된거에 따라 문법 해석

인코더 1=b 3=c이런거

incoder 암호화(딥러닝에서 인코딩 디코딩이 있음)

핵심적 특징을 추출해서

바로가기 지정해주는것이 환경변수

시스템에 쓰이는 전역변수라고 생각하면 됨.

에디터 (편집기) ide(통합개발장비)

비주얼코드는 에디터이다.(ide가 아니라)

비주얼코드랑 vs코드랑 다름

vs코드는 커스텀해서 쓸수 있음(확장코드가 다 되어있어서 사용하기 추천함)

쌌고수들은 메모장에다가 코딩함.

play 버튼 누르면??(x)

터미널에다가 쉘에다가 파이썬 경로를 직접써서 하는게 정석임(

ssh. 네트워크

vs코드는 원래 메모장임→ 내 입맛에 맞게 custom



amd??(회사)뭔뜻

64bit용 명령어셋(요즘)

32bit에서 불량나는 경우 있음(cpu)

프로그램(갯앰프드)→32bit →amd최신cpu돌리니까 꺼짐



모니터 키보드 바꾸면 갑자기 잘됨

비싼 메인보드는 다 쉴드 처리가 되어있음

객체사이에 느린부분을 c로 바꾸어서

(오후수업)



★네트워크 개념

서버설치어떻게 함??

서버는 원격을 통해 쓰고 있음(지금)ssh 프로토콜이용해서 씬

★SSH-원격통신규격

ssh:프로토콜, 프로토콜이란?(규약, 약속)

통신 (커뮤니케이션)

telnet

www.(world wide web)

htt

p(통신규약 즉 프로토콜)://



))주소:(ip address) Host라고도 함. 주인————(사설ip, 공인ip)

도로명(V4)-32bit

192.168.0.1(4자리라서 V4)

지번명(V6)-118bit

6자리

사설ip 옆동네 순이집(우리동네만 호환,내부에서만 호환)

공인ip 구체적인 번지(어디서나 다 통용되는)→공유기 번호와 연관(주소똑같지만 위치는 다름)) 아이피 주소 같지만 사는 곳 다르듯이

공인 ip는 국가가 관리함.

<http://www.naver.com>(도메인 주소)라고 함.

<http://www.navar.com>(접속하지 말라고 공문낸 적있었음)

DNS(domain name server)→비싼 공유기가 알아서 해줌

도메인은 누가팔지? 산사람이 가지고 있음.소유권자가 등록되어있음(특허권처럼)

DDNS (dynamic domain name server) 철수가 이사를 갔으니 미희네 옆집으로 메핑을 해야지(너무 잘알려져서 해킹을 잘당함)

사설공유기를 쓰는데도 외부에서 접속이 가능함 hk-it (도메인 이름)

6303번 포트번호(port forwarding)→포트 전달해달라

(스타크래프트)

TCP-택배요 여기요 (카톡서버) 최신은 tcp를 쓴다

UDP-택배요(떨지고 가버림)

엠넵 (불법)

공유기 고정ip가 공인ip임

))대포폰 의 작동원리

))port : 선착장A,B,C,D,E

항구, 8가닥 RJ45(하드웨어) →물리적인 선, port 인터넷 창 여러개 띄우는 것

항구에 선착장 여러개(선착장(port)마다 역할 다 다름→well known 500번까지는 정해져있음, 포트마다 프로토콜이 정해져 있음. 특수한 역할을 수행할 수 있는 포트)

hostname은 별명

))21.telnet.FTP(텔넷과 ftp는 쉘에 접속할수있는 프로토콜임)21번 거의 안씀

))22.ssh.SFTP(보안이걸린

ssh는 명령어를 주고받을 수 있는것, 데이터도 주고받을 수 있음.

gui(x) cli(command line interface)

vnc 프로토콜(보안안됨,사용비추)

))80.http(요즘 거의안씀)

))443.https

))453 SMB

))패키지(데이터 조합)



SSH(프로토콜)

- **SSH(Secure Shell):** 원격으로 다른 컴퓨터에 접속할 때 사용하는 **보안 통신 프로토콜**입니다. 예를 들어, 내가 다른 컴퓨터에 안전하게 로그인해서 명령어를 실행하거나 파일을 주고받고 싶을 때 SSH를 사용합니다.
 - **비유:** 마치 집에 들어갈 때 사용하는 열쇠가 비밀번호로 잠긴 도어락처럼, 다른 컴퓨터에 들어갈 때 비밀번호와 암호화를 통해 보안을 지켜주는 '열쇠' 역할을 합니다.

프로토콜(규약, 약속)

- **프로토콜:** 컴퓨터끼리 데이터를 주고받을 때 **정해진 규칙**입니다. 모든 컴퓨터가 이 규칙을 따라야 서로를 이해하고 데이터를 주고받을 수 있습니다.
 - **비유:** 서로 다른 언어를 사용하는 사람들이 소통하기 위해 ****통일된 언어(규약)****로 대화를 나누는 것과 같습니다. 예를 들어, 국제 회의에서 영어를 사용해 서로 이해하는 것처럼, 컴퓨터도 프로토콜을 통해 서로 대화합니다.



vs코드 이용할때 확장자 ssh(모바일 스템, 푸티도)이용함.



접속 가능한 환경

리눅스

파이썬 3일이면한다고??

리눅스는 커널이름



우분투 재단에서 만든 리눅스 커널

파이썬

cli

맥은 리닉스

리눅스 커널(굽지않아도되는 모놀로식 구조)——센트오에스는 커널의 종류는(통합형구조, 모듈식 구조) 리눅스는 모듈식 구조
모놀로식은 커널이 한개 드라이버도 하나로 묶여있음장점:: 함부로 건드릴수 없고 빠르다. 하나로 통합되니까
모듈형구조는 패 바뀌치기 하듯이 끄지않아도 맥도 리눅스도 서버시스템에 굉장히 최적화가 되어있음 개발환경들이 리눅스에 굉장히 발달되어있음(오픈소스라서 지적재산권 모두 공유해야 한다..)

리눅스 마스터 자격증(따기 어려움)—>센트 os red hat(빨간 모자) 리눅스의 모든 기능을 다 암기해야 함.

리눅스

안드로이드는 플레이스토어 애플은 앱스토어

centos- rpm, csh
ubuntu -apt, bash

oracle도 돈내야함.

mac은 z-shell쓰

우분투 띄우면 user @ server(컴퓨터이름):——dash shell

유저는 바꿀수 있음. 관리자 권한으로 실행=(sudo) super user(임시적으로 관리자 권한을 줌)

super user su라고 치면 root#으로 바뀜

리눅스는 tree구조

리눅스 /home/user/file

윈도우는 역슬래시

/(HDD)(SDD) root(뿌리라고 함)

파일이동 명령어를 알아야함

상대경로위치와 절대경로위치(root에서 시작함)를 알아야함.

a/file/

상대경로 단축어-항상 내 위치를 기준으로 함(터미널)

.(현재 내 위치) ..(상위 디렉터리)

~home(내home) —는 뒤로 가기(홈페이지 뒤로가기랑 비슷)

change directory(CD)

명령어

cd -r-v a b

cp -r-v a b(a에서 b로 카피)

디렉터리 구조 보는 명령어

ls(list)의 줄임말

ll /home/a

ll(내 현재 디렉터리)

mkdir

rmdir (파일 지우는 거 단점: 빈 폴더 지우기만 가능→ recursive)

rm -rf al__

sudo rm-rfv /* (파일들이 다 날라간적있음)

폴더 만들고 파일 삭제하고



`mv /home/a.txt/home/b.txt`(같은 디렉토리 안에서 옮길수 있을까??-)이름 바꾸기가 됨)

`cp -r`(하위폴더까지 다 복사가 됨)

옛컴퓨터에 마우스가 없었음

`nano a.txt` 가 있었음

`vim a.txt`(씩은물)



`nvidia-smi`

`watch -n 1 v nvidia-smi`



`grep`

`ls-l`

| `grep s`(|앞에서 뒤로)

명령어는 다 있다.



centos

-apt, rpm, snap, dpkg

`sudo apt update`

`sudo apt upgrade`

`apt remove` 패키지이름

`purge *`(미친놈이 다지움)

`install`

`install -f` 파괴된 패키지 알아서 복구

