**Linux** – eshte system operativ me I perhapur per serveret ne internet. Ai eshte FREE si dhe Multi-user dhe Multi-tasking.

**Multi-User** – dmth qe resurset e sistemit operativ(rami,procesori…etj) ndahen ne menyre te drejte ndermjet shfrytezueseve.

**MultiTasking** – Shume procese ekzekutohen njekohesisht

Komponentet kryesore te Unix/LINUX jane:

1. KERNEL
2. SHELL
3. PROGRAMET

**GREP**- eshte komande me ane te te cilies mund te bejm kerkimin e shprehjeve ne kuader te nje fajlli tekstual.

**Process** – quhet programi ne ekzekutim dhe rrethina ne te cilen ai ekzekutohet.

Cdo process e ka numrin e vet identifikues te quajtur **PID** (process Id)

Preceset komunikojne permes sinjaleve. Nje process mund te nderprehet me komenden **KILL.**

**Root –** eshte administrator I sistemit ne Linux

Ekzistojne dy lloje te variableve: lokale dhe globale.

Variablat lokale- ekzistojne vetem ne shell

Variablat globale – ekzistojne edhe ne rrethinen e te gjitha programeve

Strings - jane vektor i char.

Ndryshe quhen edhe lidhje e shenjave qe perfundon me ‘\0’

**Pointeret -** jane variabla qe ruajne adresat memorike ten je variable tjeter.Ata mundesojne call-by reference.

Deklarimi i pointereve behet ne kete menyre: int \*myptr

Ja nje shembyll:

Int y = 5;

Int \*yptr;

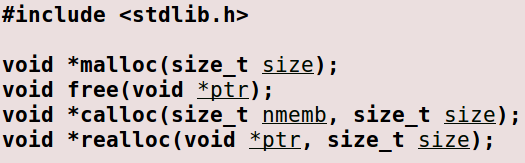
yptr= &y; -- inicializimi I pointerit

**Operatori &** - Përdorët për ta lexuar adresën memorike të një variable

Pointeret na ndihmojne ta perdorim HEAP MEMORY (alokimin dinamik te memories).

void \*malloc - rezervon memorien

void free – e liron memorin



**Serializimi** nënkutpon procesin e konvertimit të gjendjës së një objekti në një format të ruajtjes së të dhënave, p.sh. në disk, apo në rrjetë

**Deserializimi** nënkutpon procesin rikthimit të gjendjës së objektit të serializuar nga formati i ruajtjës së të dhënave, p.sh. nga disku, apo nga hostet tjera në rrjetë

Sfidat e serializimit jane:

* Byte Order (Endianness)
  + Renditja e byte-ave gjatë reprezentimit binar të numrave, para së gjithash int
  + Makina të ndryshme përdorin Endianness të ndryshëm
  + Problemi lind kur makina me Endianness jo kompatibël lexojnë shënimet e shkruara nga makina tjetër
* Endianness më të përhapur:
  + **Big-Endian:** 
    - motorola-format ndryshe njihet si ***network byte order***
    - ***Power-PC, Raspberry Pi, etc.***
  + **Little-Endian:** 
    - intel-format shumë i përhapur në domainin e PC-ve.
* Serializimi/Deserializimi arrihet përmes sys-calls:
  + ***hton*** gjatë shkruarjes/transmetimit të variablave int & Co. në disk/rrjetë
  + ***ntoh*** gjatë leximit/interpretimit të variablave int & Co. nga disku/rrjeta

**Proces** – quhet nje program ne ekzekutim

Procesi perbehet nga (ID,Variabla,Staku)

* **Multitasking** - Aftësia e një sistemi operativ për të ekzekutuar më shumë një program njekohësisht.
* ***Time-sharing***
  + Resorset lirohen në bazë kohore, ose në bazë të prioriteteve.
  + Në kete menyre një applikacioni nuk mund ti bllokoj resorset kur ato nevojiten nga applikacioni tjetër.
* Sjellja“ (behaviour) i një procesi karakterizohet me ane të listës së komandave të ekzekutuara të një procesi
* Një liste e tille quhet ***trace***

**Gjendjet e procesit –**

* *Ready*
  + Procesi është i gatshem për ekzekutim dhe pret për tu ekzekutuar nga CPU.
* *Blocked*
  + Procesi pret për një event (ngjarje) (p.sh. perfundimi i një operacioni I/O) – procesi nuk është në gjendje për tu ekzekutuar

*Running*

* + Procesi është në posedim të CPU dhe është në ekzekutim

**Cka eshte Swapping?**

* *Swapping* = shkarkimi i proceseve në një memorje sekundare (p.sh. Harddisku)
* Për realizim të kesaj në OS na nevojiten 2 „gjendje“ dhe dy „queue“ të reja
  + *Ready, suspend*

*Blocked, suspend*

* Numri unik i procesit – **PID** (proc. identifier) është Indeks në tabelen primare të proceseve (Primary Process Table)
* Identifikimi i shfrytezuesit (User identifier)
  + Shfrytezuesi të cili i takon procesi
  + Numri i procesit të cilin e ka gjeneruar procesin (Parent Process Identifier) – PPID
* Për ti mbrojtur shtrukturat e të dhenave të OS, ekzistojne dy mode të ekzekutimit
  + Modi i previligjuar (system mode, kernel mode, supervisor mode, control mode)  
    përdoret për qasje në regjistrat kontrollues, instruksione primitive të I/O, MM
  + Modi i shfrytezuesit (User mode)
* *Mode Switch* në mes të *User/Kernel Mode*
* *Process Switch* behët nderpreja e ekzekutimit të procesit për shkak të *interaptave*, *gabimeve* ose *Supervisor Call*