### 08/04/2025

## 1. Linux Nedir?

**Linux**, bilgisayarın donanımıyla yazılımların iletişim kurmasını sağlayan **bir çekirdektir (kernel)**. Yani, işletim sisteminin en temel parçasıdır.

Ama insanlar çoğu zaman "Linux" derken sadece çekirdeği değil, çekirdekle birlikte gelen **tam işletim sistemini** kastederler. Bu tam işletim sistemleri genelde açık kaynaklıdır ve ücretsizdir. Örneğin:

* **Ubuntu**
* **Debian**
* **CentOS**

Bu sistemler, Linux çekirdeği üzerine inşa edilmişlerdir. Yani özünde hepsi "Linux tabanlı"dır ama her biri farklı görünümler, araçlar ve amaçlarla geliştirilmiştir.

**Neden Önemli?:** Bilimsel hesaplama ve özellikle moleküler dinamik simülasyonları için Linux sıkça tercih edilir çünkü:

* **Performans ve Kararlılık:** Sunucu ve HPC (Yüksek Performanslı Hesaplama) ortamları için çok güvenilir ve stabildir.
* **Esneklik ve Kontrol:** Komut satırı arayüzü (CLI), sistem üzerinde detaylı kontrol ve otomasyon imkanı sunar. Simülasyonları başlatmak, izlemek, yönetmek ve sonuçları işlemek için komut satırı vazgeçilmezdir.
* **Açık Kaynak:** Geniş bir topluluk tarafından desteklenir, birçok bilimsel yazılım öncelikle Linux için geliştirilir veya en iyi performansı Linux üzerinde verir.
* **Maliyet:** Genellikle ücretsizdir.

## 2. Temel Kavramlar

* **Terminal (Uçbirim):** Linux ile etkileşim kuracağınız metin tabanlı arayüz. Komutları buraya yazarsınız.
* **Shell (Kabuk):** Komutlarınızı yorumlayan ve işletim sistemine ileten program (örneğin Bash, Zsh).
* **Komut Yapısı:** Genellikle komut [seçenekler] [argümanlar] şeklindedir.
  + komut: Çalıştırılacak programın adı (örn: ls, cd).
  + [seçenekler]: Komutun davranışını değiştiren genellikle - veya -- ile başlayan parametreler (örn: ls -l).
  + [argümanlar]: Komutun üzerinde işlem yapacağı şeyler, genellikle dosya veya dizin adları (örn: cp dosya1.txt hedef\_dizin/).
* **Dosya Sistemi Hiyerarşisi:**
  + /: Kök dizin, her şeyin başlangıcı.
  + ~: Ev dizininiz (/home/kullanici\_adiniz).
  + .: Bulunduğunuz mevcut dizin.
  + ..: Bir üst dizin.

## 3. Temel Linux Komutları ve Kullanımları

### **Chatgpt4.5**

#### **Dosya ve Dizin İşlemleri**

1. pwd: Bulunduğunuz dizinin tam yolunu gösterir.
2. ls: Dizin içeriğini listeler.  
   * Örnek: ls -l (Detaylı listeleme)
3. cd: Dizin değiştirme komutu.  
   * Örnek: cd dizin\_adı
   * Bir üst dizine çıkma: cd ..
4. mkdir: Yeni dizin oluşturur.  
   * Örnek: mkdir yeni\_dizin
5. rmdir: Boş dizini siler.  
   * Örnek: rmdir eski\_dizin
6. touch: Yeni dosya oluşturur veya dosya zaman damgasını günceller.  
   * Örnek: touch dosya.txt
7. cp: Dosya veya dizin kopyalar.  
   * Örnek: cp kaynak.txt hedef.txt
   * Tüm dizini kopyalama: cp -r kaynak\_dizin hedef\_dizin
8. mv: Dosya veya dizin taşır ya da yeniden adlandırır.  
   * Örnek: mv eski\_ad.txt yeni\_ad.txt
9. rm: Dosya siler.  
   * Örnek: rm dosya.txt
   * Tüm dizini içeriği ile silme: rm -r dizin\_adı

#### **Dosya İçeriğini Görüntüleme ve Düzenleme**

1. cat: Dosya içeriğini görüntüler.  
   * Örnek: cat dosya.txt
2. less: Dosyayı sayfa sayfa gösterir.  
   * Örnek: less dosya.txt
3. nano: Basit bir metin editörüdür.  
   * Örnek: nano dosya.txt
4. vim: Gelişmiş bir metin editörüdür.  
   * Örnek: vim dosya.txt

#### **Sistem Bilgisi ve Kontrol Komutları**

1. man: Komutların kullanım kılavuzlarını gösterir.  
   * Örnek: man ls
2. --help: Bir komutun desteklediği seçenekleri gösterir.  
   * Örnek: ls --help
3. sudo: Yönetici yetkisiyle komut çalıştırır.  
   * Örnek: sudo apt update
4. top: CPU kullanımını canlı gösterir.
5. htop: Gelişmiş sistem kaynakları izleyicisi.
6. nvidia-smi: GPU kullanımı ve doluluk bilgisini gösterir (NVIDIA kartlar için).

#### **Kullanıcı ve İzin İşlemleri**

1. chmod: Dosya/dizin izinlerini değiştirir.  
   * Örnek: chmod 755 dosya.txt
   * Eger bir dosya calistirilabilir degilse: chmod +x dosya.txt
2. chown: Dosya sahibini değiştirir.  
   * Örnek: sudo chown kullanıcı dosya.txt
3. passwd: Kullanıcı şifresini değiştirir.

#### **Arama ve Sorgulama**

1. grep: Metin içinde arama yapar.  
   * Örnek: grep 'kelime' dosya.txt
2. find: Dosya veya dizin arar.  
   * Örnek: find . -name "\*.txt"

locate’de hızlı dosya arar (önceden index gerekir)

* + Örnek: locate config.yaml

#### **Dosya Transferleri ve Arşivleme**

1. scp: Uzak sunucuya dosya kopyalar.  
   * Örnek: scp dosya.txt user@server:/home/user
2. wget: İnternetten dosya indirir.  
   * Örnek: wget http://example.com/dosya.txt
3. tar: Dosyaları arşivleyip sıkıştırır veya açar.  
   * Sıkıştırma: tar -czvf arşiv.tar.gz dizin
   * Açma: tar -xzvf arşiv.tar.gz

#### **Disk ve Hafıza Bilgileri**

1. df: Disk kullanım bilgisini verir.
2. du: Dizin veya dosya boyutunu gösterir.  
   * Örnek: du -sh dizin
3. free: RAM kullanımını gösterir.

#### **Ağ İşlemleri**

1. ping: Ağ bağlantısını test eder.
2. ifconfig / ip a: Ağ ara yüz bilgilerini gösterir.
3. ssh: Uzak sunucuya bağlanır.  
   * Örnek: ssh kullanıcı@ip\_adresi

#### **Zamanlama ve İş Yönetimi**

1. cron: Otomatik görev zamanlama aracı.
2. jobs: Arka plandaki işleri listeler.
3. bg: İşleri arka plana gönderir.
4. fg: İşleri ön plana getirir.

#### **Diğer Önemli Komutlar**

1. history: Kullanılan son komutları listeler.
2. clear: Terminal ekranını temizler.
3. alias: Komut kısayolu oluşturur.  
   * Örnek: alias ll="ls -alF"
4. env: Ortam değişkenlerini gösterir.
5. echo: Metni ekrana basar.

#### **Temel Linux Komutları için Yardım**

* Genel yardım almak için:  
  + man komut\_adı
  + komut\_adı --help

#### **Örnek CPU ve GPU Kontrol Komutları**

* CPU izleme: top veya htop
* GPU izleme (NVIDIA): nvidia-smi

### **Gemini2.5 Pro**

**3.1. Yardım Alma**

1. **man [komut]**: Komutlar hakkında detaylı kullanım kılavuzunu (manual) gösterir. İçinde gezinmek için ok tuşları, sayfalar arası geçiş için Boşluk (ileri) ve b (geri), çıkmak için q tuşunu kullanın.
   * man ls (ls komutunun kılavuzunu gösterir)
2. **[komut] --help** veya **[komut] -h**: Çoğu komut, kendi kullanımı hakkında kısa bir yardım metni sunar.
   * cp --help (cp komutunun yardım metnini gösterir)
   * ssh -h (Bazen -h daha kısa bir özet verir)

**3.2. Dizinlerde Gezinme**

1. **pwd**: (Print Working Directory) Mevcut bulunduğunuz dizinin tam yolunu gösterir.
   * pwd
2. **cd [dizin\_yolu]**: (Change Directory) Belirtilen dizine geçiş yapar.
   * cd /home/kullanici/projeler (Belirtilen tam yola gider)
   * cd belgelerim (Mevcut dizin içindeki 'belgelerim' dizinine gider)
   * cd ~ (Ev dizininize gider)
   * cd .. (Bir üst dizine çıkar)
   * cd - (Bir önceki bulunduğunuz dizine döner)
3. **ls [seçenekler] [dizin]**: (List) Belirtilen dizinin içeriğini (dosya ve alt dizinleri) listeler. Belirtilmezse mevcut dizini listeler.
   * ls (Mevcut dizini listeler)
   * ls /etc (/etc dizinini listeler)
   * ls -l (Detaylı liste formatı: izinler, sahip, boyut, tarih)
   * ls -a (Gizli dosyalar dahil (. ile başlayanlar) tüm dosyaları gösterir)
   * ls -lh (Detaylı liste ve dosya boyutlarını okunabilir formatta (KB, MB, GB) gösterir)
   * ls -lt (Zamana göre (en yeni üstte) sıralı listeler)
   * ls -ltr (Zamana göre ters sıralı (en eski üstte) listeler)

**3.3. Dosya ve Dizin Yönetimi**

1. **mkdir [dizin\_adi]**: (Make Directory) Yeni bir dizin oluşturur.
   * mkdir simulasyon\_girdileri
   * mkdir -p projeler/analiz/grafikler (-p ile iç içe dizinleri tek seferde oluşturur, var olanları atlar)
2. **rmdir [dizin\_adi]**: (Remove Directory) Boş bir dizini siler.
   * rmdir eski\_veriler (Dizin boş değilse hata verir)
3. **touch [dosya\_adi]**: Boş bir dosya oluşturur veya var olan bir dosyanın son erişim/değiştirme zamanını günceller.
   * touch input.mdp
4. **cp [kaynak] [hedef]**: (Copy) Dosya veya dizinleri kopyalar.
   * cp girdi.txt girdi\_yedek.txt (Dosyayı aynı dizinde farklı isimle kopyalar)
   * cp girdi.xtc ../analiz/ (Dosyayı bir üst dizindeki 'analiz' klasörüne kopyalar)
   * cp -r proje\_A/ proje\_B/ (-r ile dizinleri ve içeriklerini kopyalar. Hedef dizin yoksa oluşturur)
5. **mv [kaynak] [hedef]**: (Move) Dosya veya dizinleri taşır veya yeniden adlandırır.
   * mv eski\_isim.log yeni\_isim.log (Dosyayı yeniden adlandırır)
   * mv sonuclar.dat ../arsiv/ (Dosyayı bir üst dizindeki 'arsiv' klasörüne taşır)
   * mv proje\_X/\* . (proje\_X içindeki tüm dosyaları mevcut dizine taşır)
6. **rm [seçenekler] [dosya\_adi]**: (Remove) Dosyaları siler. **DİKKAT: Silinen dosyalar genellikle geri getirilemez!**
   * rm gecici\_dosya.tmp (Dosyayı siler)
   * rm -i \*.log (-i ile silmeden önce onay ister)
   * rm -r eski\_proje/ (-r ile dizini ve içindekileri siler. Çok dikkatli kullanın!)
   * rm -rf cok\_eski\_proje/ (-f ile sormadan siler. **ÇOK TEHLİKELİ!** Ne yaptığınızdan emin değilseniz kullanmayın.)

**3.4. Dosya İçeriğini Görüntüleme**

1. **cat [dosya\_adi]**: (Concatenate) Dosyanın tüm içeriğini ekrana basar. Kısa dosyalar için uygundur.
   * cat parametreler.txt
   * cat dosya1.txt dosya2.txt > birlesik\_dosya.txt (Dosyaları birleştirip yeni dosyaya yazar)
2. **less [dosya\_adi]**: Uzun dosyaları sayfa sayfa görüntülemek için kullanılır. man komutu gibi gezinilir (ok tuşları, Boşluk, b, / ile arama, q ile çıkış).
   * less simulasyon\_ciktisi.log
3. **more [dosya\_adi]**: less gibidir ama daha eski ve daha az özelliğe sahiptir. Genellikle less tercih edilir.
   * more buyuk\_veri.csv
4. **head [seçenekler] [dosya\_adi]**: Dosyanın başlangıç kısmını (varsayılan olarak ilk 10 satırı) gösterir.
   * head trajectory.pdb
   * head -n 20 script.py (İlk 20 satırı gösterir)
5. **tail [seçenekler] [dosya\_adi]**: Dosyanın son kısmını (varsayılan olarak son 10 satırı) gösterir. Log dosyalarını veya devam eden simülasyon çıktılarını takip etmek için kullanışlıdır.
   * tail sonuclar.log
   * tail -n 50 analiz\_raporu.txt (Son 50 satırı gösterir)
   * tail -f simulasyon.log (-f ile dosya büyüdükçe yeni eklenen satırları sürekli gösterir. Ctrl+C ile çıkılır)

**3.5. Dosya İçeriğinde Arama**

1. **grep [seçenekler] [aranan\_ifade] [dosya\_adi/dizin]**: (Global Regular Expression Print) Dosya(lar) içinde belirli bir metin veya deseni arar ve eşleşen satırları gösterir.
   * grep "Error" simulasyon.log (log dosyasında "Error" geçen satırları bulur)
   * grep -i "atom" structure.pdb (-i ile büyük/küçük harf duyarsız arama yapar)
   * grep -r "Energy" . (-r ile mevcut dizin ve alt dizinlerdeki tüm dosyalarda "Energy" arar)
   * grep -v "Warning" input.log (-v ile "Warning" geçmeyen satırları gösterir)
   * grep -n "Total" report.txt (-n ile eşleşen satırların numaralarını da gösterir)
2. **find [nerede] [ne\_aranacak] [ne\_yapilacak]**: Dosya sisteminde belirli kriterlere göre dosya veya dizin arar.
   * find . -name "\*.pdb" (Mevcut dizin ve altında adı .pdb ile biten dosyaları bulur)
   * find /data -type d -name "sim\_\*" (/data altında adı sim\_ ile başlayan dizinleri bulur)
   * find ~ -mtime -7 (Ev dizininizde son 7 gün içinde değiştirilmiş dosyaları bulur)
   * find . -name "\*.log" -exec rm {} \; (Bulunan .log dosyalarını siler. **Dikkatli kullanın!**)

**3.6. Dosya İzinleri ve Sahiplik**

1. **chmod [izinler] [dosya/dizin]**: (Change Mode) Dosya veya dizinlerin erişim izinlerini (okuma r, yazma w, çalıştırma x) değiştirir.
   * chmod +x script.sh (script.sh dosyasına çalıştırma izni ekler)
   * chmod 644 veri.txt (Sahibine okuma/yazma (6), grup ve diğerlerine sadece okuma (4) izni verir)
   * chmod -R 755 proje\_dizini/ (-R ile dizin içindekilere de aynı izni uygular. 755: Sahip rwx, grup rx, diğerleri rx)
2. **chown [sahip]:[grup] [dosya/dizin]**: (Change Owner) Dosya veya dizinin sahibini ve/veya grubunu değiştirir. Genellikle yönetici (root) yetkisi gerekir.
   * sudo chown yeni\_kullanici:lab\_grubu analiz.log (sudo komutu yönetici hakları ile çalıştırmayı sağlar)
3. **sudo [komut]**: (Super User Do) Komutu yönetici (root) yetkileriyle çalıştırır. Şifrenizi girmeniz istenir. Yazılım yükleme, sistem ayarları değiştirme gibi işlemler için gereklidir. **Gereksiz yere kullanmaktan kaçının.**

**3.7. İşlem (Process) Yönetimi**

1. **ps [seçenekler]**: (Process Status) Çalışan işlemleri listeler.
   * ps (O anki terminaldeki işlemlerinizi gösterir)
   * ps aux (Sistemdeki tüm kullanıcıların tüm işlemlerini detaylı gösterir)
   * ps -ef (Farklı formatta tüm işlemleri gösterir)
   * ps aux | grep gromacs (Çalışan gromacs işlemlerini bulmak için grep ile birlikte kullanılır)
2. **top**: Sistem kaynak kullanımını (CPU, Bellek) ve çalışan işlemleri anlık olarak gösteren interaktif bir araçtır. q ile çıkılır.
3. **htop**: top komutunun daha renkli, kullanıcı dostu ve interaktif bir versiyonudur. Genellikle ayrıca yüklenmesi gerekir (sudo apt install htop veya sudo yum install htop). q ile çıkılır.
4. **kill [PID]**: Belirli bir işlem numarasına (PID - Process ID) sahip işlemi sonlandırmak için sinyal gönderir (varsayılan sinyal TERM'dir, nazikçe kapatmayı dener). PID numarasını ps veya top/htop ile bulabilirsiniz.
   * kill 12345
5. **kill -9 [PID]**: İşlemi zorla kapatır (KILL sinyali). Sadece normal kill işe yaramadığında kullanılmalıdır, veri kaybına neden olabilir.
   * kill -9 12345
6. **Ctrl+C**: Terminalde ön planda (foreground) çalışan bir işlemi durdurur/iptal eder.
7. **Ctrl+Z**: Terminalde ön planda çalışan bir işlemi askıya alır (suspend) ve arka plana atar. İşlem durur ama bitmez.
8. **bg**: (Background) Ctrl+Z ile askıya alınan son işlemi arka planda çalışmaya devam ettirir.
9. **fg**: (Foreground) Arka planda çalışan veya askıya alınmış bir işlemi tekrar ön plana getirir.

**3.8. Sistem Bilgisi ve Kaynak İzleme**

1. **uname [seçenekler]**: (Unix Name) Sistem bilgilerini gösterir.
   * uname -a (Tüm bilgileri gösterir: Kernel adı, sürümü, makine mimarisi vb.)
2. **df [seçenekler]**: (Disk Free) Disk kullanımını gösterir.
   * df -h (-h ile boyutları okunabilir formatta gösterir)
3. **du [seçenekler] [dizin/dosya]**: (Disk Usage) Belirtilen dosya veya dizinin ne kadar disk alanı kapladığını gösterir.
   * du -sh . (-s özet, -h okunabilir formatta mevcut dizinin toplam boyutunu gösterir)
   * du -h --max-depth=1 (Mevcut dizindeki dosya ve alt dizinlerin boyutlarını ayrı ayrı gösterir)
4. **free [seçenekler]**: Sistemdeki boş ve kullanılan bellek (RAM) ve takas (swap) alanını gösterir.
   * free -h (-h ile okunabilir formatta gösterir)
5. **lscpu**: CPU (İşlemci) mimarisi hakkında detaylı bilgi verir (çekirdek sayısı, model vb.).
6. **nvidia-smi**: (NVIDIA System Management Interface) Eğer sistemde NVIDIA ekran kartı varsa, GPU kullanımı, bellek kullanımı, sıcaklık gibi bilgileri anlık olarak gösterir. MD simülasyonları için GPU kullanımını takip etmek çok önemlidir.
7. **uptime**: Sistemin ne kadar süredir açık olduğunu ve ortalama yükünü gösterir.

**3.9. Ağ Komutları**

1. **ping [adres]**: Belirli bir sunucuya veya IP adresine ağ paketleri gönderip yanıt alarak bağlantıyı test eder. Ctrl+C ile durdurulur.
   * ping google.com
2. **ssh [kullanici]@[sunucu\_adresi]**: (Secure Shell) Başka bir Linux sunucusuna güvenli bağlantı kurarak komut çalıştırmanızı sağlar. HPC kümelerine bağlanmak için temel komuttur.
   * ssh kullaniciadim@cluster.universite.edu.tr
3. **scp [seçenekler] [kaynak] [hedef]**: (Secure Copy) ssh üzerinden dosyaları güvenli bir şekilde iki makine arasında kopyalar.
   * scp yerel\_dosya.txt kullanici@sunucu:/hedef/dizin/ (Yerelden sunucuya kopyalama)
   * scp kullanici@sunucu:/kaynak/dosya.dat . (Sunucudan yerel makineye kopyalama)
   * scp -r yerel\_dizin/ kullanici@sunucu:/hedef/ (-r ile dizin kopyalama)

**3.10. Arşivleme ve Sıkıştırma**

1. **tar [seçenekler] [arsiv\_adi.tar] [dosyalar/dizinler]**: (Tape Archive) Dosyaları tek bir arşiv dosyasında birleştirir veya arşivden çıkarır. Genellikle sıkıştırma ile birlikte kullanılır.
   * tar -cvf proje\_arsivi.tar proje\_dizini/ (create, verbose-göster, file. Arşiv oluşturur)
   * tar -xvf proje\_arsivi.tar (xtract. Arşivi açar)
   * tar -czvf proje\_arsivi.tar.gz proje\_dizini/ (z ile gzip sıkıştırması yapar)
   * tar -xzvf proje\_arsivi.tar.gz (gzip ile sıkıştırılmış arşivi açar)
   * tar -cjvf proje\_arsivi.tar.bz2 proje\_dizini/ (j ile bzip2 sıkıştırması yapar)
   * tar -xjvf proje\_arsivi.tar.bz2 (bzip2 ile sıkıştırılmış arşivi açar)
2. **gzip [dosya\_adi]**: Dosyayı .gz uzantısıyla sıkıştırır (orijinal dosyayı siler).
   * gzip buyuk\_veri.csv (sonuç: buyuk\_veri.csv.gz)
3. **gunzip [dosya\_adi.gz]**: .gz ile sıkıştırılmış dosyayı açar (orijinal .gz dosyasını siler).
   * gunzip buyuk\_veri.csv.gz (sonuç: buyuk\_veri.csv)

**3.11. Giriş/Çıkış Yönlendirme ve Boru Hattı (Pipe)**

1. **>**: Komutun standart çıktısını (ekrana yazılacak olanı) bir dosyaya yönlendirir (dosya varsa üzerine yazar).
   * ls -l > dosya\_listesi.txt
2. **>>**: Komutun standart çıktısını bir dosyanın sonuna ekler (dosya yoksa oluşturur).
   * echo "Simulasyon Bitti: $(date)" >> durum.log
3. **|**: (Pipe - Boru Hattı) Bir komutun çıktısını başka bir komutun girdisi olarak bağlar. Komutları zincirlemek için çok güçlü bir yöntemdir.
   * ps aux | grep mdrun (Tüm işlemler listelenir, çıktı grep'e gönderilir ve sadece mdrun içeren satırlar gösterilir)
   * history | grep cd (Komut geçmişinde cd içeren komutları bulur)

**3.12. Diğer Kullanışlı Komutlar**

1. **echo [metin]**: Verilen metni veya değişkenin değerini ekrana yazar.
   * echo "Merhaba Dünya"
   * echo $PATH (PATH çevre değişkeninin değerini gösterir)
2. **history**: Daha önce kullandığınız komutların listesini gösterir.
3. **clear**: Terminal ekranını temizler.
4. **whoami**: Mevcut giriş yapmış kullanıcının adını gösterir.
5. **passwd**: Kullanıcı şifrenizi değiştirmenizi sağlar.

**4. CPU ve GPU Kullanımını Kontrol Etme**

* **CPU:**
  + top veya htop komutlarını kullanın. %CPU sütunu her bir işlemin CPU kullanım yüzdesini gösterir. Üst kısımdaki özet bilgi sistemin genel CPU kullanımını verir.
  + lscpu komutu ile işlemcinizin kaç çekirdekli olduğunu ve diğer özelliklerini öğrenebilirsiniz.
* **GPU (NVIDIA Kartlar için):**
  + nvidia-smi komutunu çalıştırın. Bu komut size:
    - GPU modelini ve sürücü sürümünü
    - GPU'nun anlık kullanım yüzdesini (GPU-Util)
    - GPU belleğinin ne kadarının kullanıldığını (Memory-Usage)
    - GPU sıcaklığını
    - GPU üzerinde çalışan işlemleri (Compute processes) gösterir.
  + Simülasyonlarınızın GPU'yu verimli kullanıp kullanmadığını kontrol etmek için bu komut çok değerlidir. watch nvidia-smi komutu ile çıktıyı belirli aralıklarla otomatik olarak yenileyebilirsiniz (Ctrl+C ile çıkılır).

**5. Örnek Bir Çalışma Akışı**

1. Yeni bir proje dizini oluştur: mkdir ~/md\_projem
2. O dizine git: cd ~/md\_projem
3. Gerekli girdi dosyalarını (örneğin başka bir yerden) kopyala: cp /data/ortak\_alan/gromacs\_inputs/\* .
4. Dosyaları listele: ls -lh
5. Parametre dosyasına göz at: less input.mdp (q ile çık)
6. Simülasyonu başlat (örnek komut, gerçek komutunuz farklı olacaktır): gmx\_mpi mdrun -deffnm simulation\_run > md.log &
   * > : Çıktıyı md.log dosyasına yönlendirir.
   * & : İşlemi arka planda başlatır, terminali kullanmaya devam edebilirsiniz.
7. Çalışan işlemleri kontrol et: ps aux | grep mdrun
8. GPU kullanımını kontrol et: nvidia-smi
9. CPU ve genel sistem durumunu izle: htop (q ile çık)
10. Simülasyon ilerlemesini log dosyasının sonundan takip et: tail -f md.log (Ctrl+C ile çık)
11. Simülasyon bitince (veya durdurulunca) sonuçları incele: ls \*.xtc \*.gro \*.log
12. Sonuçları arşivle: tar -czvf ../md\_projem\_sonuclar.tar.gz .