



**Yıldız Technical University**  
**Computer Engineering**  
**2023-2024 Spring**  
**BLM2022 Computer Organization**  
**Homework 4**

Soru 1)

CPU modelinizi <https://www.computerhope.com/issues/ch000046.htm> adresinde verilen şekilde belirleyiniz. Cevabınıza CPU modeline dair ekran çıktısını da ekleyiniz.

Linux için aşağıdaki komut da kullanılabilir:

\$ lscpu

<https://www.cpu-world.com/> adresinden CPU modelinizi arayarak, CPU'nun farklı seviyelerdeki veri ve komut önbellek (data and instruction cache) boyutlarını ve haritalama tiplerini not ediniz. Cevabınıza cache boyutlarına ve haritalama tiplerine dair ekran çıktısını da ekleyiniz.

Bundan sonraki işlemleri sanal makine üzerinde ubuntu kurulumu ile veya mevcut ise fiziki ubuntu kurulumu üzerinden devam ediniz.

gcc kurulumunu gerçekleyiniz.

\$ sudo apt install build-essential

valgrind kurulumunu gerçekleyiniz.

\$ sudo apt-get install valgrind

Ekte verilen 1.c, 2.c ve 3.c dosyalarında matris çarpımına ilişkin farklı sıradaki döngü değişkenlerinin kullanımı ile oluşan hafıza ve komut adreslemeden doğan cache erişim durumlarını inceleyiniz.

Bunun için öncelikle kod derlemek için aşağıdaki adımları izleyiniz.

\$ gcc 1.c -o 1.out

\$ gcc 2.c -o 2.out

\$ gcc 3.c -o 3.out

Oluşan çalıştırılabilir dosyaları valgrind'in cachegrind aracını kullanarak analiz ediniz.

\$ valgrind --tool=cachegrind ./1.out

\$ valgrind --tool=cachegrind ./2.out

\$ valgrind --tool=cachegrind ./3.out

CPU'nun özelliklerini inceleyerek bulduğunuz cache boyut ve haritalama tiplerini dikkate alarak, boş bir cache ile başlandığı durumda, yürütülen kodlar için hem veri hem de komut açısından **L1 seviye** cahe hit ve miss oranlarını teorik olarak hesaplayınız. Elde ettiğiniz teorik sonuçları valgrind çıktıları ile karşılaştırınız.

İşlemleri kod içindeki **dimension** sabiti 64 ve 256 değerlerindeyken tekrarlayınız.

**Kodları çalıştırarak gerçekleşen matris çarpım işlem sürelerini not ediniz. Oluşan süreler ile cache hit miss oranlarını karşılaştırarak değerlendiriniz.**