## Dağıtık Sistemler 1. Hafta: Dağıtık Sistemlere Giriş

📌 **Teorik İçerik:**

* Dağıtık sistem nedir?
* Merkezi, dağıtık ve paralel sistemler arasındaki farklar
* Dağıtık sistemlerin avantajları ve zorlukları
* Gerçek dünya örnekleri: Google, Amazon, Facebook, Netflix

🖥️ **Laboratuvar:**  
✅ Linux’ta temel terminal komutları (SSH, SCP, rsync)  
✅ Basit bir **istemci-sunucu modeli** Python ile uygulama  
✅ Python’da **socket** programlama ile basit TCP bağlantısı

**2. Hafta: Dağıtık Sistem Mimarileri**

📌 **Teorik İçerik:**

* Monolitik, Mikroservis ve SOA Mimarileri
* P2P (Eşler Arası) ve Merkezi Mimariler
* Katmanlı ve Paylaşımlı Mimariler
* Veri Dağıtımı: Replikasyon, Fragmentasyon

🖥️ **Laboratuvar:**  
✅ Docker ile **basit bir dağıtık servis** çalıştırma  
✅ Nginx kullanarak **Load Balancing** yapılandırma  
✅ Basit bir **gRPC servisi** geliştirme

**3-4. Hafta: Paralel ve Dağıtık Hesaplama**

📌 **Teorik İçerik:**

* Paralel ve dağıtık hesaplama farkları
* Paralelleştirmenin temelleri: Task Parallelism ve Data Parallelism
* Dağıtık algoritmaların sınıflandırılması
* Moore Yasası ve Amdahl Yasası
* Paralel Programlama Modelleri: **Paylaşımlı Bellek, Mesaj Geçişli (MPI)**
* SIMD, MIMD, SPMD modelleri
* GPU vs. CPU hesaplama

🖥️ **Laboratuvar:**  
✅ Python **multiprocessing** kütüphanesi ile çoklu işlem uygulamaları  
✅ Bash script ile **çoklu işlem (fork, exec)** uygulaması  
✅ Linux **htop** ile CPU çekirdek kullanımını gözlemleme

✅ Python’da **NumPy ile vektör işlemleri** (SIMD örneği)  
✅ OpenMP için **temel programlama ortamı kurulumu**  
✅ Çoklu çekirdek işlemi OpenMP ile hızlandırma

**5. Hafta: OpenMP**

📌 **Teorik İçerik:**

* OpenMP temel komutları
* Fork-Join modeli
* OpenMP direktifleri (#pragma omp parallel)
* Hata ayıklama ve performans ölçümü

🖥️ **Laboratuvar:**  
✅ OpenMP ile **"Hello World"** programı  
✅ Paralel matris çarpımı uygulaması  
✅ OpenMP ile **kritik bölgeler (critical sections) ve deadlock örnekleri**

**6. Hafta: MPI (Message Passing Interface)**

📌 **Teorik İçerik:**

* MPI'nin temel özellikleri
* Noktadan Noktaya (Point-to-Point) ve Kolektif (Collective) İletişim
* Senkron ve asenkron mesajlaşma
* MPI'de hata yönetimi

🖥️ **Laboratuvar:**  
✅ MPI ortamı kurulumu ve mpirun komutları  
✅ MPI ile **"Hello World"** programı  
✅ MPI ile **paralel matris çarpımı**

**7. Hafta: İsimlendirme**

📌 **Teorik İçerik:**

* Dağıtık sistemlerde adlandırma türleri
* DNS ve merkezi isimlendirme sistemleri
* Konsistent Hashing

🖥️ **Laboratuvar:**  
✅ Python’da **kendi DNS sistemimizi** oluşturma  
✅ **Etcd ve Zookeeper** kullanarak dağıtık yapılandırma

**8. Hafta: Senkronizasyon**

📌 **Teorik İçerik:**

* Mutex, Semaphore, Spinlock
* Raft ve Paxos Algoritmaları
* Koordinasyon problemleri

🖥️ **Laboratuvar:**  
✅ Python’da **semafor ve mutex ile yarış koşullarını engelleme**  
✅ Zookeeper ile dağıtık senkronizasyon

**9. Hafta: Tutarlılık**

📌 **Teorik İçerik:**

* ACID ve BASE modelleri
* Tutarlılık protokolleri (Eventual, Strong, Causal)
* CAP Teoremi

🖥️ **Laboratuvar:**  
✅ Redis ve Cassandra ile **Eventual Consistency testi**  
✅ Amazon DynamoDB **replikasyon deneyleri**

**10. Hafta: Hata Toleransı ve Güvenlik**

📌 **Teorik İçerik:**

* **Hata Toleransı**
  + Çökmeye karşı dayanıklılık
  + Leader Election (Raft, Paxos)
  + Dağıtık hata yönetimi (Timeout, Checkpointing)
* **Güvenlik**
  + Kimlik doğrulama (Authentication)
  + Yetkilendirme (Authorization)
  + TLS/SSL ile güvenli iletişim
  + Veri bütünlüğü ve şifreleme

🖥️ **Laboratuvar:**  
✅ RabbitMQ ile mesaj kuyruğunda hata toleransı  
✅ OpenSSL ile **veri şifreleme ve sertifikalar**  
✅ JWT tabanlı kimlik doğrulama uygulaması

**11-12. Hafta: Blockchain ve Dağıtık Defter Teknolojisi**

📌 **Teorik İçerik:**

* **Blockchain Nedir?**
  + Merkezi olmayan yapı
  + Konsensüs algoritmaları (Proof of Work, Proof of Stake, PBFT)
  + Akıllı sözleşmeler ve dApp’ler
* **Blockchain’in Dağıtık Sistemlerle İlişkisi**
  + Dağıtık güvenlik modeli
  + Byzantine Hata Toleransı
  + Veri değişmezliği (immutability)
* **Gerçek Hayat Kullanım Alanları**
  + Kripto paralar (Bitcoin, Ethereum)
  + Tedarik zinciri yönetimi
  + Dijital kimlik doğrulama

🖥️ **Laboratuvar:**  
✅ Python ile **basit bir blockchain** oluşturma  
✅ Solidity ile **akıllı sözleşme geliştirme**  
✅ Ethereum ağına **test kontratı dağıtma (Remix, Metamask)**  
✅ Hyperledger Fabric kurulumu ve basit bir blockchain ağı oluşturma

## ****13-14. Hafta: Proje Sunumları****

📌 **Her öğrenci, Blockchain ilgili bir proje geliştirecek ve sunacaktır.**