

## BLG 503 PARALEL PROGRAMLAMA

## DERSİN AMACI

- Dersin amacı:
  - paralel programlamada karşılaşılan temel problemleri anlamak
  - paralel programlamayı destekleyen dillerin bu problemlere sağladıkları çözümleri öğrenmek

## ARDIŞIL PROGRAMLAMA

- Ardışıl programlama bir “**ardışıl algoritma**”nın yürütülmesini öngörür.
- Ardışıl algoritma, **her adımın bir işleme** karşı düştüğü bir adımlar dizisi tanımlar.
- Ardışıl algoritmalar, temel olarak işlemleri ardışıl bir sırada yürüten geleneksel bilgisayar mimarileri için uygundur.
- Bu tür mimariler ilerleyen teknoloji ile ne kadar hızlansalar da, gereksinimlere yanıt veremeyebilirler.

## PARALEL PROGRAMLAMA

- Hız problemine çözüm olarak paralel bilgisayar mimarileri kullanılabilir.
- Paralel bilgisayarlar bir adımda çok sayıda işlem yürütürler.
- Bir problemi paralel bilgisayar üzerinde çözebilmek için, **her adımı için çoklu işlemler tanımlayan bir algoritmaya-paralel algoritma**-gerek duyarız.

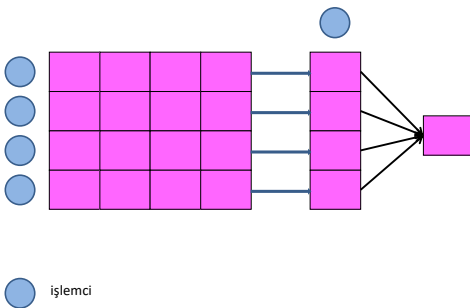
## ÖRNEK

- Problem:
  - $n \times n$  elemanlı bir matrisin elemanlarının toplamını hesapla
- Ardışıl çözüm: tüm elemanlara bir kere erişerek toplamı hesapla
- Sonuca  $n^2$  adımda ulaşılır

## ÖRNEK

- Paralel çözüm: iki aşamadan oluşur
- 1. aşama:
  - Matrisin her satırında yer alan elemanların paralel bir şekilde toplayalım
  - Bu adım sonunda  $n$  elemanlı bir dizi oluşur.
- 2. aşama:
  - İlk aşama sonunda oluşan dizi elemanlarını topla

## ÖRNEK



## PARALEL PROGRAM

- **Paralel algoritma:** *her adımı için çoklu işlemler*
- **Paralel program:** Belirli bir problemin çözümü için, paralel yürütülmekte olan ardışıl çoklu işlemler topluluğu.
- Çoklu işlem topluluğu: proses, iplik, görev(task)
- Her birinin kendine ait bir kontrol akışı vardır.
- Üzerinde çalışılan mimariye bağlı olarak gerçek veya lojik (zaman paylaşım) paralel yürütme

## PROSES ETKİLEŞİMİ

- Paralel programı oluşturan prosesler ortak bir hedef için, etkileşim içinde, birarada çalışırlar.
- Etkileşim:
  - **Haberleşme:** birbirleri ile bilgi alış veriş
  - **Senkronizasyon:** çalışmanın sürdürülmesi bir başka prosesin denetimine bağlı olduğu koşullar
  - **Karşılıklı dışlama:** ortak bellek alanlarına erişimin denetlenmesi

## PARALEL PROGRAMLAMA YAPILARI

- Proses etkileşimi için önerilen çözümler iki farklı modelde incelenebilir:
  1. Paylaşılan Bellek: proseslerin ortaklaşa erişebildikleri paylaşılan bir bellek alanı mevcuttur
  2. Mesaj Aktarımı: Ortak bir bellek alanı yoktur, prosesler bir mesaj aktarım ortamı aracılığı ile etkileşimde bulunurlar.
- Farklı modeller için farklı çözümler üretilmiştir

## DERSİN AMACI

- Paralel programlar geliştirirken karşılaşılan problemleri anlamak
- Bu problemler için geliştirilmiş olan çözüm yaklaşımlarını öğrenmek
- Paralel program geliştirme desteği sunan programlama dillerinin sağladıkları yapıları incelemek

## KONULAR

- Derste işlenecek konular:
  1. Paylaşılan Bellek Modeli
    - Meşgul beklemeli çözümler
    - Meşgul bekleme gerektirmeyen çözüm (semafor)
    - Monitör yapısı (Concurrent Pascal)
  2. Mesajlaşma Altyapısına Dayalı Çözümler
    1. CSP
    2. Occam
    3. ADA

### DERS KİTABI

- Dersin bir bölümünde izlenen kitap:
  - “Principles of Concurrent and Distributed Programming” -M. Ben-Ari
- CSP, OCCAM ve ADA için makaleler verilecektir

### DEĞERLENDİRME

- Arasınnav : %40 (12 Nisan 2020)
- **ÖNEMLİ: ARASINAVIN TELAFİSİ YOKTUR**
- Final : %40
- Proje : %20
- **Vize Koşulu:** Proje teslim etmiş olmak