

Ders 1:

- Algoritma: bir problemi çözmek, bir işi yapmak için gerekli adımların listesi
- Yemek tarifi:
 - 10 dakika ısıt (for)
 - Pembeleşinceye kadar ısıt (while)
 - Evde kekik yoksa nane koy (if / else)
 - Süzgeçten tencereye aktar (atama)
- Yol tarifi:
 - X'i görünceye kadar düz git (while)
 - 100 metre sonra sola dön (for)
- Robota tahtaya çember çizdirin
 - Bilgisayarlar söylediğimizi yapar, istediğimizi değil
 - Ve söylediklerimizi çok net bir dille söylemeliyiz.
 - Bilgisayar dilleri bunun için var
- Amacımız: Algoritma geliştirmek değil, iyi algoritma geliştirmek
- İyi algoritma
 - Daha doğru: hava tahmini, satranç motorları, video öneri sistemi-bir deneyim
 - Daha hızlı: mergesort
 - Daha az bellek tutan: quicksort
 - Yukarıdaki kriterlerin hiç biri her zaman en önemli değil. Çok daha doğru ama 100 kat yavaş çalışan bir algoritma? Ödünleşim (trade-off)
 - Sizin programınızla birlikte aynı kaynakları kullanan başka programlarda var
- İyi algoritma geliştirmek genelde tek seferde olmaz. İteratif bir süreç.
- Algoritma geliştirmek nasıl öğrenilir:
 - Kimse öğretmez, sadece yardımcı olabilir
 - Algoritma geliştirmek, algoritma geliştirerek öğrenilir.
 - Ne yazık ki algoritma geliştirmenin bir algoritması yok ☹
- Örnek A: Gauss ilkokulda, 1'den N'e kadar ki sayıları toplayın
 - N çift ise $N/2$ tane $N+1$ olur
 - N tek ise $(N-1)/2$ tane $N+1$ ve ortadaki $(N+1)/2$
 - İkisi de $N*(N+1)/2$, N işlem yerine sabit 3 işlem
- Örnek B: $M > N$ olmak üzere $-N$ ile $+M$ arası sayıları toplayalım
 - Klasik yol: $(N+M)$ işlem
 - $-N + N$ arası toplam 0. $N+1$ ile M arası kalır. $(M-N)$ işlem
 - $1-M$ toplamını $1-N$ toplamından çıkar. Sabit 7 işlem.
- Google nasıl Google oldu? Bir kullanıcı araması ile bir web sayfasının benzerliğini, web sayfasının güvenilirliği ile birleştiren ve bunu çok hızlı yapabilen bir algoritma geliştirdi (Pagerank). Önceden elle yapılan bir işlemi otomatik hale getirdi.
- İyi haber: Ve daha iyileştirilmeyi / çözülmeyi bekleyen bir sürü çözüm / problem var ☺