

Algoritma ve Problem Çözme İlişkisi

DR. H. GÖKÇE BİLGİÇ DOĞAN

BTAE 102 – ALGORİTMA TASARIMI VE GELİŞTİRME

HAFTA 3

Algoritma nedir?

- Günlük hayatta algoritmayı nasıl kullanırız?
 - Bir öğrencinin sabah yataktan kalkıp okula gidene kadar yaptığı tüm sıralı işlemler
 - Kek yaparken takip edilen tüm işlem adımları
 - Çay demlerken takip edilen tüm işlem adımları
- Başka aklımıza gelen neler var?

Günlük hayattan örnekleri inceleyelim:

Bir öğrencinin yataktan kalkıp işe gidene kadar olan tüm sıralı işlem adımları

- Yataktan kalk
- Elini yüzünü yıka
- Kahvaltını yap
- Dişlerini fırçala
- Pijamalarını çıkar
- Okul kıyafetlerini giy
- Okul çantayı hazırla
- Servise bin
- Okula git

Günlük hayattan örnekleri inceleyelim: Çay demlemek için sıralı işlem adımları

- Çaydanlığa su doldur
- Çaydanlığı ocağa koy
- Ocağın altını aç
- Suyun kaynamasını bekle
- Su kaynadı ise demliğe çay koy
- Kaynayan suyla demliği doldur
- Çaydanlığın boşalan kısmını suyla doldur
- Demlik ve çaydanlığı ocağa koy
- Suyun kaynamasını bekle

**Burada sıkıntılı gördüğünüz
bir işlem adımı var mı?**

Günlük hayattan örnekleri inceleyelim: Çay demlemek için sıralı işlem adımları

- Çaydanlığa su doldur
- Çaydanlığı ocağa koy
- Ocağın altını aç
- Suyun kaynamasını bekle
- Su kaynadı ise demliğe çay koy
- Kaynayan suyla demliği doldur
- Çaydanlığın boşalan kısmını suyla doldur
- Demlik ve çaydanlığı ocağa koy
- Suyun kaynamasını bekle

Demlik ve çaydanlığı ocağa koy



Demliği ve çaydanlığı ocağa ayrı ayrı mı koyacaksınız?

Demliği mi çaydanlığın üstüne koyacaksınız ve çaydanlığı mı demliğin üstüne koyacaksınız?

Günlük hayattan örnekleri inceleyelim: Çay demlemek için sıralı işlem adımları

- Çaydanlığa su doldur
- Çaydanlığı ocağa koy
- Ocağın altını aç
- Suyun kaynamasını bekle
- Su kaynadı ise demliğe çay koy
- Kaynayan suyla demliği doldur
- Çaydanlığın boşalan kısmını suyla doldur
- Demlik ve çaydanlığı ocağa koy
- Suyun kaynamasını bekle

Algoritmada bu gibi belirsizlikler olmamalıdır!!

Demlik ve çaydanlığı ocağa koy

Düzeltilme önerisi

Demliği ve çaydanlığın üzerine koyarak ocağa koy.

Algoritmanın temel özellikleri:

- Kesin olmalı
- Yürütülebilir olmalı
- Sıralı olmalı
- Sonlu olmalı

Algoritma nedir?

Bir problemi çözmek için takip edilecek sıralı ve sonlu sayıda işlem adımlarından oluşan çözüm yolu.

Bir görevin/işin nasıl yerine getirileceğine yönelik işlem adımlarının sıralı olarak sunulması.

Algoritma nedir?

Bir problemi çözmek için takip edilecek sıralı ve sonlu sayıda işlem adımlarından oluşan çözüm yolu.

Bir görevin/işin nasıl yerine getirileceğine yönelik işlem adımlarının sıralı olarak sunulması.

Algoritmalar...

- Açık ve net olmalı, belirsiz adımlara yer vermemeli
- Yürütülebilir olmalı
- Bir başlangıç durumu ve bitişi olmalı
- Sonlu sayıda adımdan oluşmalı
- Adımların hangi sırada gerçekleştirileceği net bir şekilde belirtilmeli
- Algoritmalarda büyük ve karmaşık olan işlemler daha küçük ve basit adımlara bölünür.

Şimdi biraz da bilgisayar ortamında problem çözüm adımlarına ilişkin tanımlara bakalım:

- Programlama
- Algoritma
- Kodlama

Programlama nedir?

- İnsanların makinelerle iletişimlerini sağlamak için uygun söz dizimlerini tasarlayarak doğru bir şekilde yazma işlemine programlama denir.
- Programlama iki alt unsurdan oluşmaktadır. Bunlar:
 - Analiz/Tasarım
 - Kodlama

Programlama nedir?

- Analiz/Tasarım -> Problem çözümü için doğru tasarlamak.
- Kodlama -> Bu tasarlanan yolu kodlarla ifade etmek.
 - Yani kodlama ile problemin çözümü için tasarlanan yol bir programlama dili kullanılarak bilgisayarın anlayacağı dile çevrilir.

Algoritma nedir?

- Dolayısıyla algoritma dediğimiz aslında analiz ve tasarım sürecinde ortaya koyduğumuz problemin çözüm yoludur.
- Diğer bir ifade ile bir problemin bilgisayar dilinde çözümünü oluştururken henüz analiz aşamasında bir programlama dilinde kodlanmadan önce programın genel işleyişinin mantıksal olarak sıralı işlem adımları ile ortaya konulmasıdır.

Algoritmalar neden önemlidir?

- Programın kodlanmasını kolaylaştırır.
- Algoritması hazırlanmış bir programı tasarlayan dışında bir başkası da kodlayabilir.
- Takım çalışmasını kolaylaştırır.
- Programdaki mantıksal hataların kodlama öncesinde tespitini kolaylaştırır.
- Kontrolü kolaylaştırır ve kodlama hatalarını azaltır.

Bilgisayar ortamında problem çözümü...

1. Problemi anlamak (İnceleme, analiz).
2. Alternatif çözüm yolları ortaya koymak.
3. Ortaya konan çözüm yollarından birini seçerek programın algoritmasını tasarlamak.
4. Tasarlanan algoritmanın bir programlama dilinde kodlamak.
5. Programın kodlandığı editörde hataların tespitini yaparak gerekli düzeltmeleri yapmak.
6. Örnek girdi ve çıktı değerleri ile programın çalışmasını test etmek.

Yazılım geliştirme süreci

1. **Analiz:** Problemin analiz edilmesi
2. **Tasarım:** En uygun çözüm yolunun bulunması
3. **Geliştirme:** Algoritmanın oluşturulması ve devamında bir programlama dilinde kodlanması
4. **Test:** Doğrulama ve hatalardan arındırma işlemlerinin yapılması
5. **Dağıtım ve Bakım:** Programın devreye alınması ile ilgili her türlü hazırlık sürecinin tamamlanması.

Algoritma örnekleri

- Çamaşır makinesinde çamaşır yıkama işlemlerine yönelik algoritma adımlarını tasarlayın.

Algoritma örnekleri

1. Makinenin kapağını aç.
2. Kirli çamaşırları makineye doldur.
3. Makinenin kapağını kapat.
4. Deterjan gözüne deterjan koy.
5. Yıkama programını seç.
6. Başlat düğmesine bas.

Algoritma örnekleri

- Klavyeden girilen üç sayının ortalamasını hesaplayarak ekrana yazdıran programın algoritmasını tasarlayın.

Algoritma örnekleri

1. Birinci sayıyı gir.
2. İkinci sayıyı gir.
3. Üçüncü sayıyı gir.
4. Sayıları topla.
5. Sayıların toplamını üçe böl.
6. Sonucu ekrana yazdır.

Algoritmaları biraz tanımaya başlayalım...

- Algoritmada yer alan her bir emir **komut**tur.
- Her algoritmanın bir başlangıcı bir de bitişi vardır.
- Algoritmalar sıralı işlem adımlarıdır.
- Algoritmalar tarafından işlenen en temel elemanlar **veri**dir.
 - Sayısal bilgiler
 - Metinsel bilgiler
 - Resimler
 - Sesler vb.

Algoritma örnekleri

- İki sayının çarpımını hesaplayan basit bir matematiksel işlemin algoritma adımlarını tasarlayın.

Algoritma örnekleri

1. Başla
2. Gir iki sayı (a, b)
3. Çarp girilen iki sayı ($\text{çarpım} = a * b$)
4. Göster çarpım sonucunu (çarpım)
5. Bitir

a ve b veri yani bir sonraki hafta ayrıntılı göreceğimiz değişkenlerimiz.

Bu örnekte a ve b sayıları programın girdileri, çarpım sonucu ise programın çıktısıdır.

Gir, Çarp ve Göster ifadeleri emirler yani komutlar olarak düşünülebilir.