

BLM212 Veri Yapıları

Giriş / Dersin Ana Hatları

Dersin Amacı

- Temel ve gelişmiş veri yapılarını tanıtmak.
- Farklı veri yapılarının implementasyonlarını ele almak ve algoritma verimliliğini zaman ve bellek açısından analiz etmek.

Ders Kitabı

- Ana Kaynak

- **Data Structures: A Pseudocode Approach with C** (2nd Ed.) (Course Technology)
Richard F. Gilberg & Behrouz A. Forouzan
-

- Yardımcı Kaynaklar

- **Veri Yapıları ve Algoritmalar** (11. Basım) (Papatya Yayıncılık)
Dr. Rıfat Çölkesen
- **C/C++ ile Veri Yapıları ve Çözümlü Uygulamalar** (2. Baskı) (Seçkin Yayıncılık)
Prof. Dr. Nejat Yumuşak ve M. Fatih Adak

Programlama Dili/Ortami

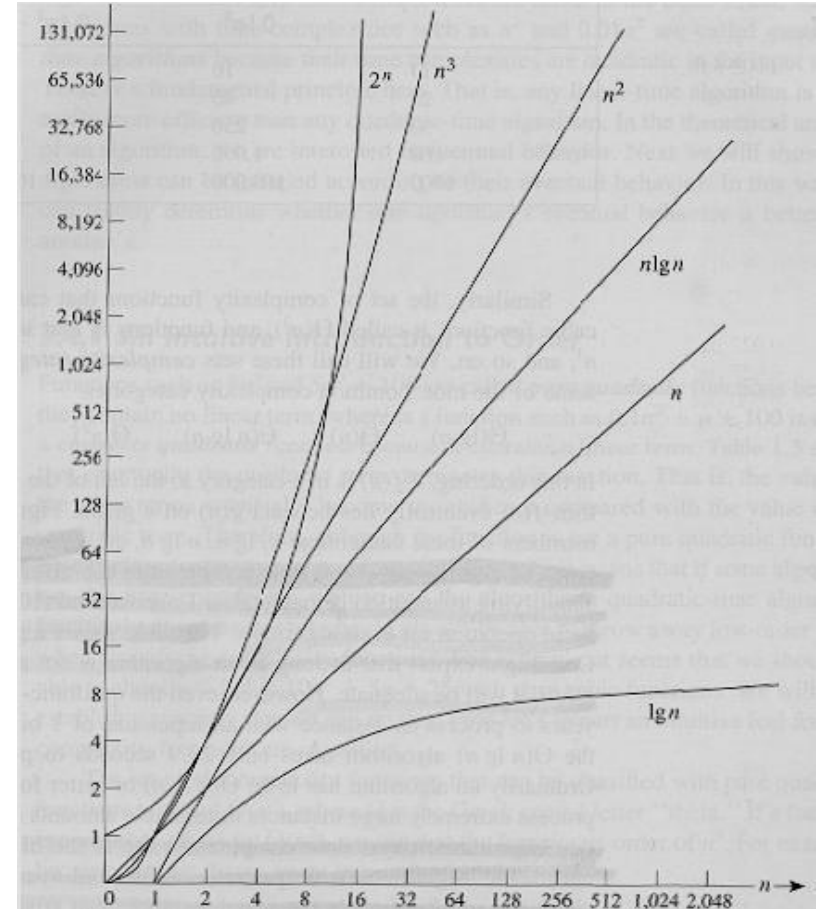
C dili / **Code::Blocks IDE**

İşlenecek Konular

- Algoritmalar ve bunların gerçekleştirilmesi
- Soyut Veri Tipi (**The Abstract Data Type-ADT**)
- Özyineleme (**Recursion**)
- Yığın ve Kuyruk (**Stacks and Queues**)
- Listeler (**Linear lists**)
- Ağaçlar (**Trees**)
- İkili Arama Ağaçları (**Binary Search Trees**)
- Heap
- Sıralama Algoritmaları
- Graflar

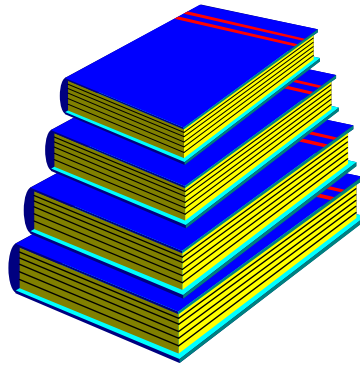
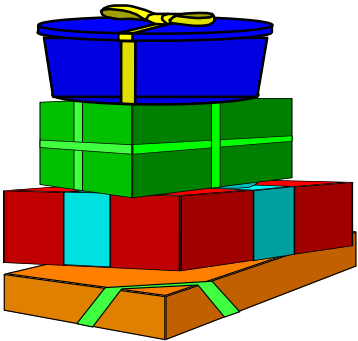
Dersin Ana Hatları

- Algoritma analizi
 - Problemin boyutu büyüdüğünde koşma zamanının (**running time**) nasıl artacağını öngörme
 - Donanımdan (kullanılan makinadan) ve programlama tarzından bağımsız olmalı
 - Best / Average / Worst

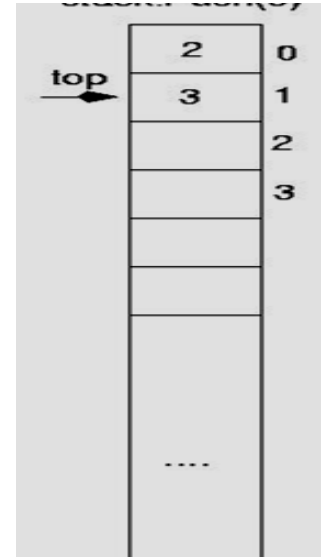


Dersin Ana Hatları

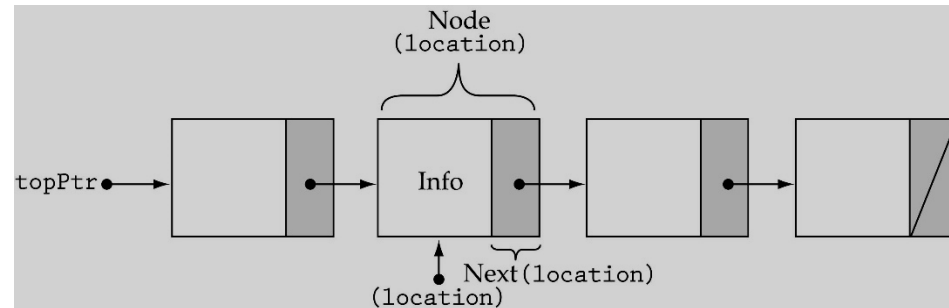
Stacks



array-based

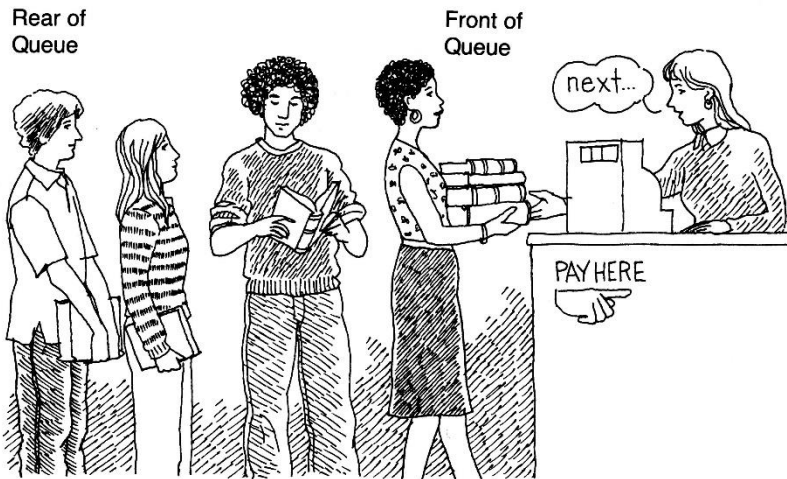


linked-list-based

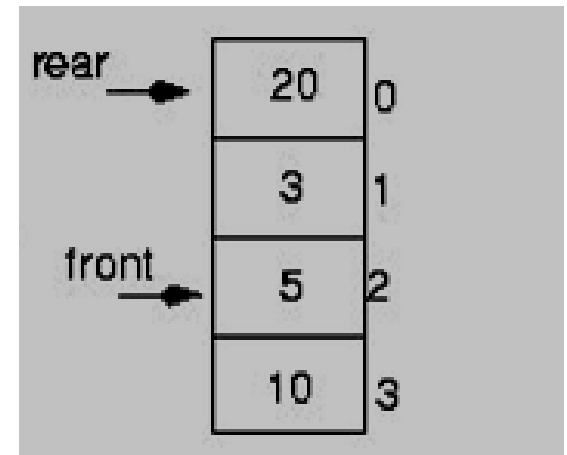


Dersin Ana Hatları

Queues



array-based



linked-list-based



Dersin Ana Hatları

- Unsorted Lists
 - array-based
- Sorted Lists
 - array-based
 - **Binary search:** very efficient search when using arrays!

Unsorted List		Sorted List	
22		12	
12		14	
46		22	
35		35	
14		46	
.		.	
.		.	
.		.	
.		.	

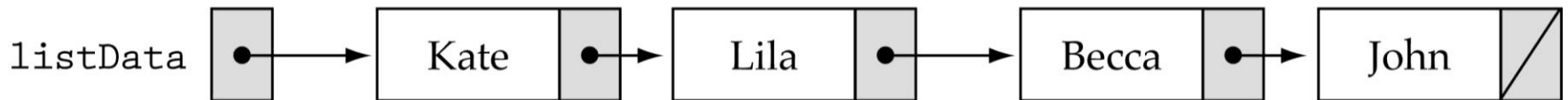
ID	Name	Address
22	John Black	120 S. Virginia Str

key

Dersin Ana Hatları

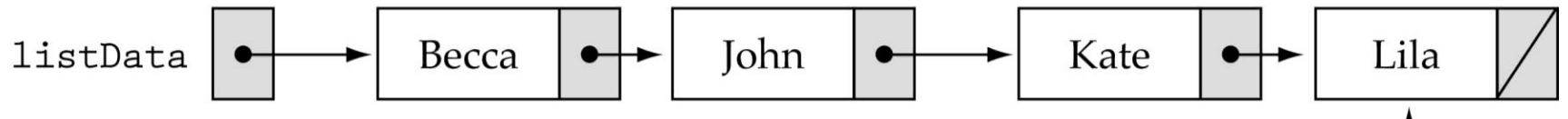
Unsorted Lists

linked-list-based



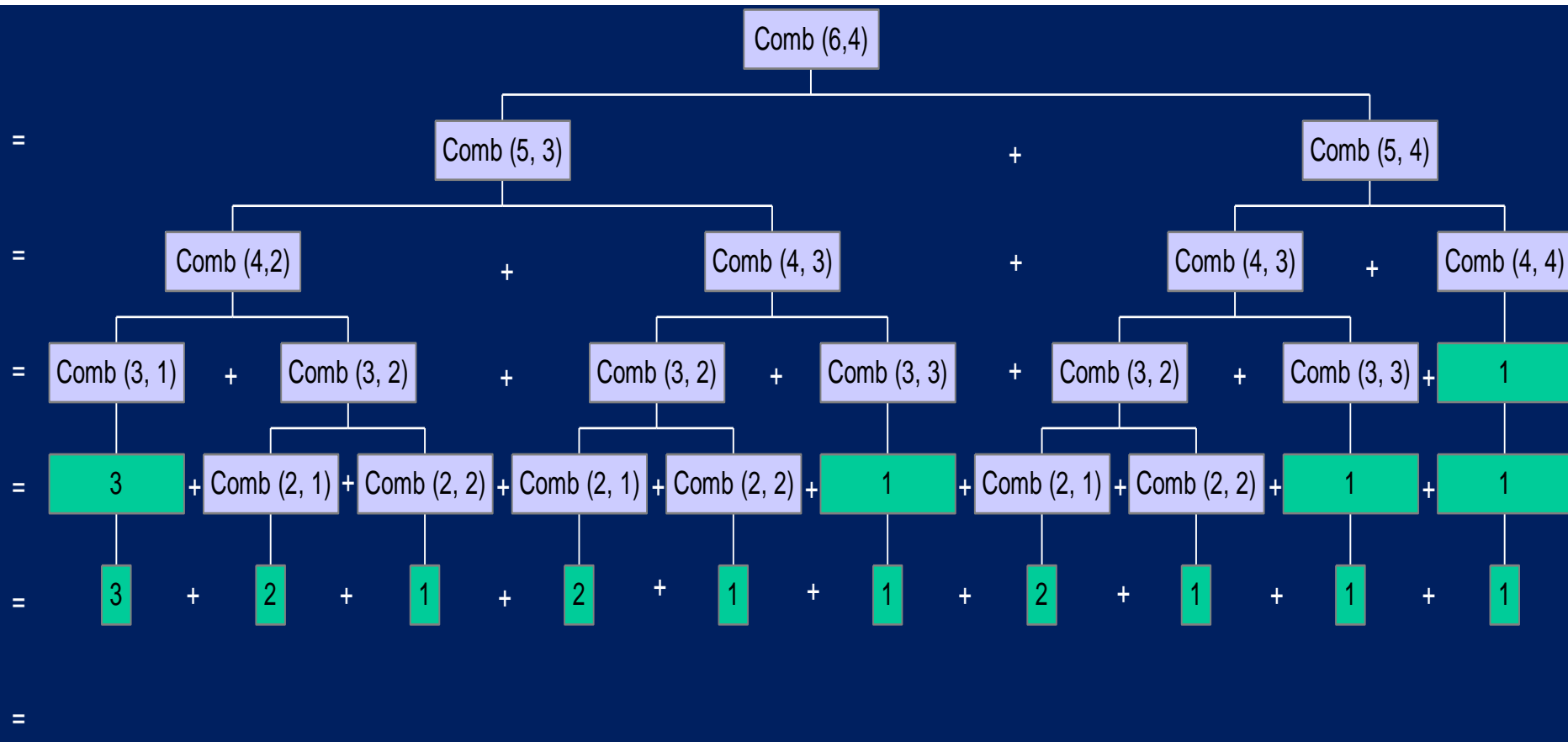
Sorted Lists

linked-list-based



Dersin Ana Hatları

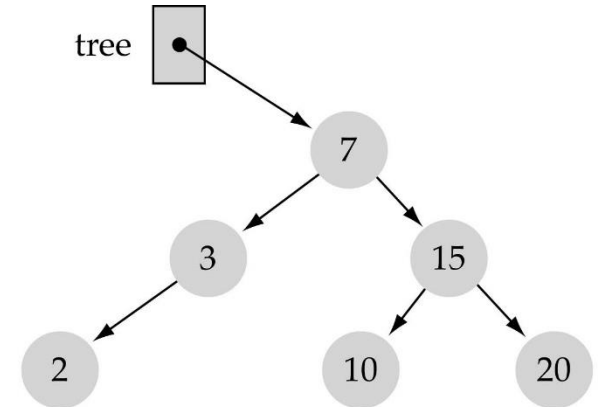
Recursion (Özyineleme)



Dersin Ana Hatları

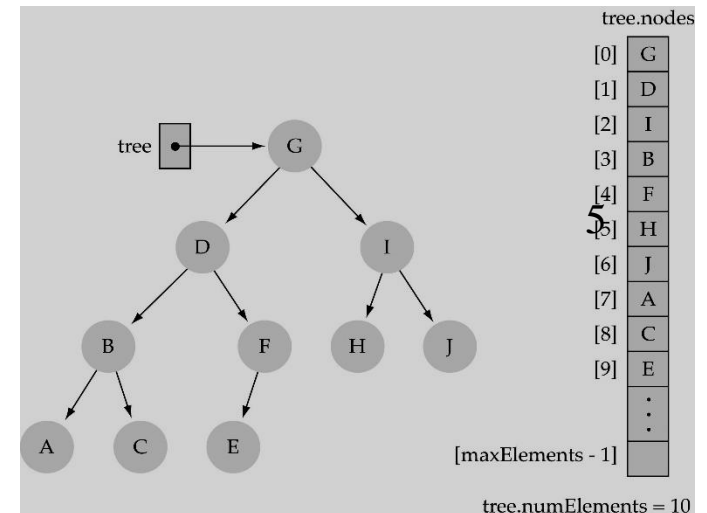
- Binary Search Trees

- Very efficient for searching!



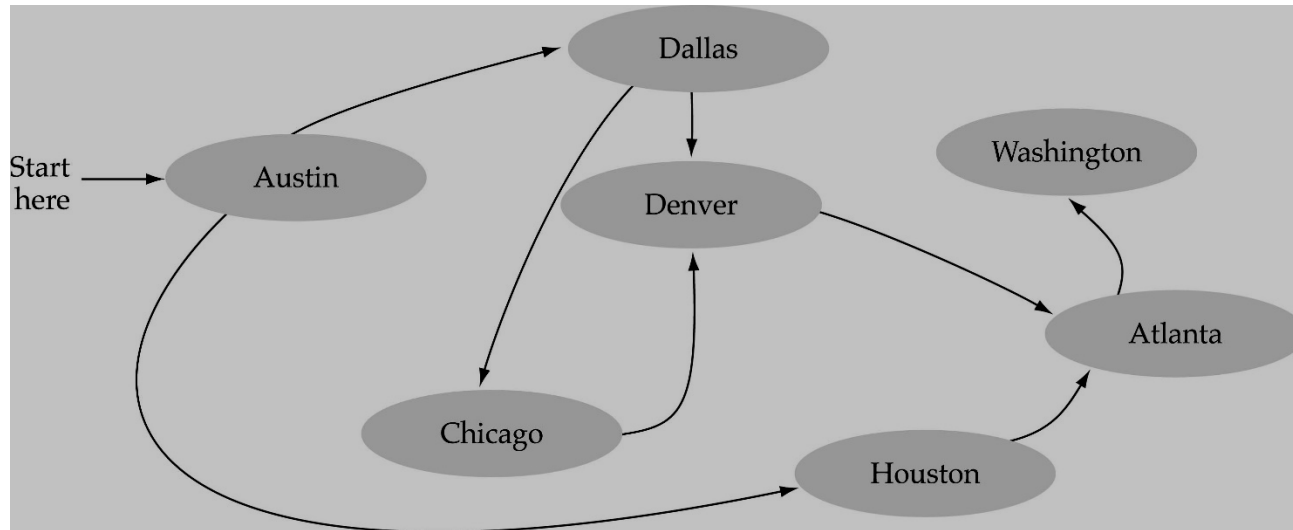
- Heaps

- Very efficient for implementing priority queues




Dersin Ana Hatları

- Graphs (i.e., array-based and linked-list-based)
 - Searching (i.e., DFS and BFS)
 - Shortest Paths (i.e., Dijkstra's and Bellman-Ford algorithms)



Dersin Ana Hatları

- Sorting
 - Selection Sort, Bubble Sort, Insertion Sort
(yavaş çalışır, fakat anlaşılması kolaydır)
 - Heapsort, Mergesort, Quicksort
(hızlıdır, fakat karmaşıktır)
 - Linear Time Sorts
(belirli koşullar altında hızlıdır)

36		6
24		10
10		12
6		24
12		36

Veri İşleme

- Herhangi bir bilgisayar kontrollü sistemde (bir bilgisayarın kendisi dahil) kontrol ve geri bildirim, farklı **verilerin analizine** ve **iletilmesine** indirgenmiştir.
- İlgili verileri analiz etmek için, matematiksel ve mantıksal açıklamada sunulabilen yöntemler kullanılır.
- **Veri işleme organizasyonu** ve **veri işleme yöntemleri** **Bilgisayar Biliminin** bir konusudur.

Algoritma

- **Algoritma**, bir problemin çözümünde izlenecek yol/talimatlar dizisi anlamına gelir.
 - Bir programlama dilinde (C, C++, Java, C# gibi) ifade edildiğinde **program** adını alır.
- Algoritma, belirli bir problemin sonucunu elde etmek için **art arda uygulanacak adımları** ve koşulları, **kesin olarak** ortaya koyar.
- Herhangi bir algoritma geliştirmek için, ilgili problemin (görevin) nasıl çözülebileceğini tam olarak bilmek gerekir.

Veri

- **Veri**, algoritmalar tarafından işlenen en temel elemanlardır (sayısal bilgiler, metinsel bilgiler, resimler, sesler ve girdi, çıktı olarak veya ara hesaplamalarda kullanılan diğer bilgiler...)
- Bir algoritmanın etkin, anlaşılır ve doğru olabilmesi için algoritmanın işleyeceği verilerin düzenlenmesi gerekir.

Veri Tipleri

TİP	UZUNLUK	DEĞER ARALIĞI
unsigned char	1 byte	0 / 255
char	1 byte	-128 / 127
enum	2 byte	-32768 / 32767
unsigned int	2 byte	0 / 65535
short int	2 byte	-32768 / 32767
int	2 byte	-32768 / 32767
unsigned long	4 byte	0 / 4.294.967.295
long	4 byte	-2.147.483.648 / 2.147.483.647
float	4 byte	$3,4 \cdot 10^{-38}$ / $3,4 \cdot 10^{38}$
double	8 byte	$1,7 \cdot 10^{-308}$ / $1,7 \cdot 10^{308}$
long double	10 byte	$3,4 \cdot 10^{-4932}$ / $1,1 \cdot 10^{4092}$

ANSI/ISO Standardına göre C dilinin veri tipleri

Veri Yapısı

- **Veri yapısı** (Data Structure) verinin veya bilginin bellekte tutulma şeklini veya düzenini gösterir.
- Verileri tanımlayan veri tiplerinin, birbirleriyle ve hafızayla ilgili tüm teknik ve algoritmik özellikleridir.
- En basit veri yapıları: **basit bir değişken** ve **bir sabit**
- Diğer veri yapıları: **diziler**, **kayıtlar**, **listeler**, **ağaçlar**, **yığınlar**, **kuyruklar** vb.

Veri Yapısı (devam...)

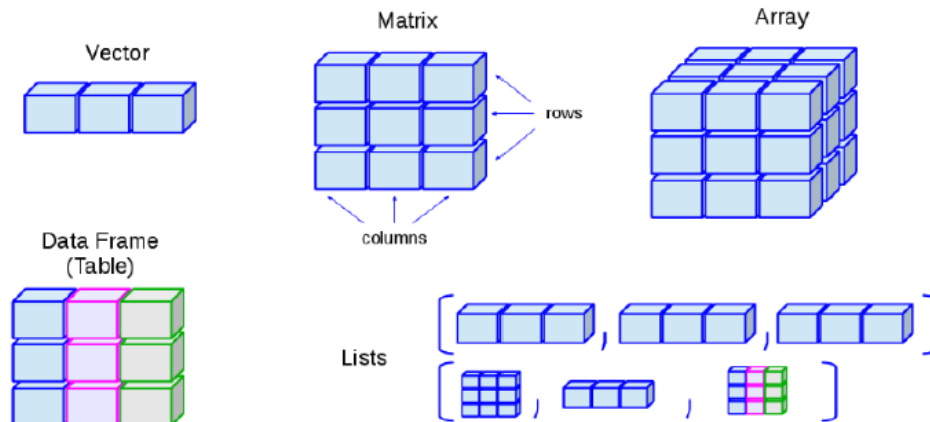
- Bilgisayar bilimlerinde işlenecek **veri miktarı** **büyüdükçe** bu verilerin

- **saklanması**,
- **sıralanması**,
- veriler içinde **arama** yapılması

gibi temel işlemlerin (hafıza, performans açısından) **verimli şekilde** gerçekleştirilebilmesi için farklı Veri Yapılarının (VY) geliştirilmesi ihtiyacı doğmuştur.

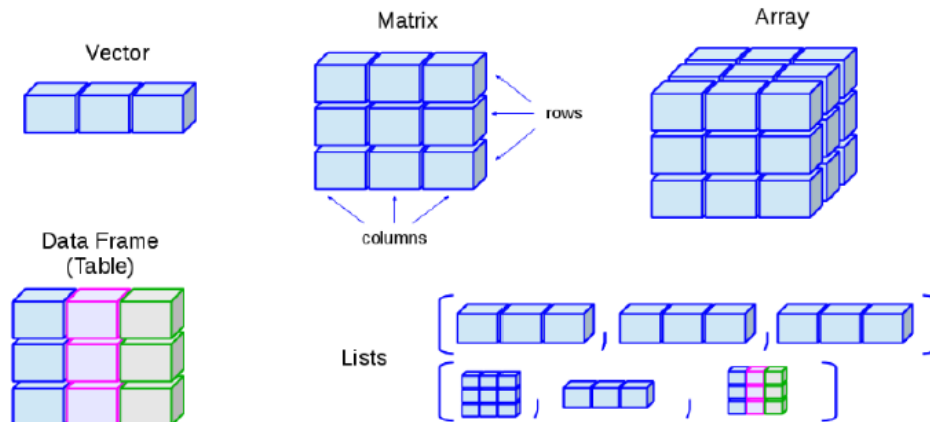
Veri Yapısı (devam...)

- Bir problemin çözümü için öncelikle **işlenecek verinin** çözüme uygun şekilde **organize** edilmesi gerekir.
- Bu bağlamda Veri Yapısı tanımı:
 - «Verinin **verimli şekilde** kullanılabilmesi için **belirli bir formatta** saklanıp **organize edilmesini** sağlayan **matematiksel modeldir.**»



Veri Yapısı (devam...)

- Bir problemin çözümü için öncelikle **işlenecek verinin** çözüme uygun şekilde **organize** edilmesi gerekir.
- Bu bağlamda Veri Yapısı tanımı:
 - «Verinin **verimli şekilde** kullanılabilmesi için **belirli bir formatta** saklanıp **organize edilmesini** sağlayan **matematiksel modeldir.**»

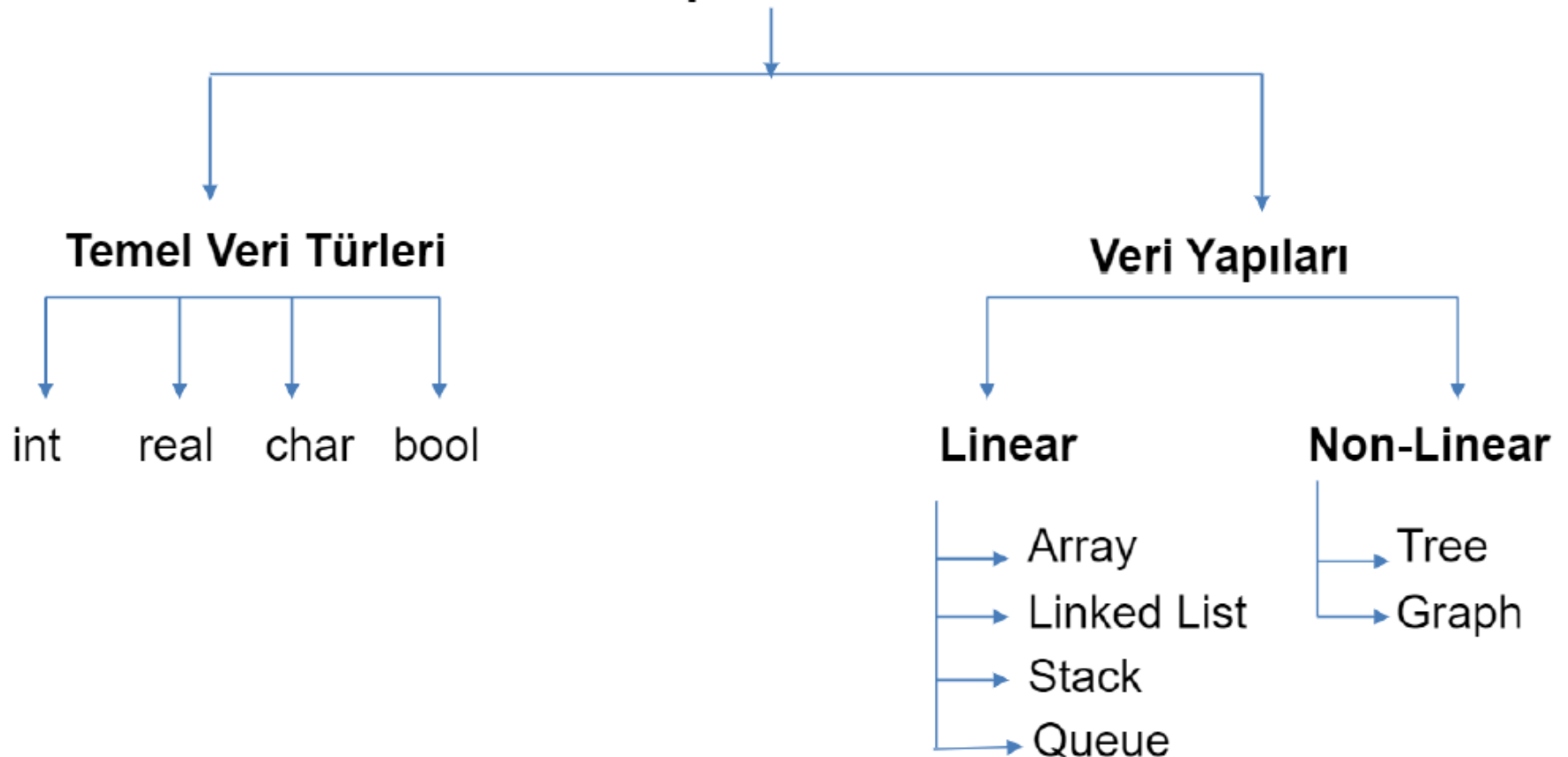


Linear ve Non-linear Veri Yapıları

- Verinin *organize edilmesine* bağlı olarak **Veri Yapıları** iki gruba ayrılabilir:
 - 1) **Linear VY**: Verinin elemanlarına *sırayla erişildiği* yapılardır.
 - Array, Linked List, Stack ve Queue gibi yapılar
 - 2) **Non-linear VY**: Verinin *lineer olmayan (hiyerarşik, network)* bir şekilde saklandığı-erişildiği yapılardır.
 - Tree ve Graph gibi

Linear ve Non-linear Veri Yapıları (devam...)

Veri Yapıları ve Türleri



Veri Yapısı_(devam...)

- Veri yapıları tüm programlama dillerinde benzer şekilde düzenlenmiştir.
 - veri yapılarının belirli bir programlama dilinden bağımsız olarak çalışılabileceği anlamına gelir.
- Veri yapılarını bilerek farklı programlama dillerini öğrenmek daha kolaydır.

İyi çalışmalar...

Kaynaklar

- Kitaplar

- **Data Structures: A Pseudocode Approach with C** (2nd Ed.) (Course Technology) Richard F. Gilberg & Behrouz A. Forouzan
- **Veri Yapıları ve Algoritmalar** (11. Basım) (Papatya Yayıncılık) Dr. Rıfat Çölkesen
- **C/C++ ile Veri Yapıları ve Çözümlü Uygulamalar** (2. Baskı) (Seçkin Yayıncılık) Prof. Dr. Nejat Yumuşak ve M. Fatih Adak

- Ders Notları

- **Veri Yapıları Ders Notları** Dr. Hakan Kutucu
- **Veri Yapıları Ders Notları** Doç. Dr. Deniz Kılınç