

Algoritmalar ve Programlama II – BLM 102

Hafta 1: Giriş



Fenerbahçe Üniversitesi

Öğretim Elemanları

Öğretim Üyesi: Dr. Vecdi Emre Levent

Ofis: 311

Email: emre.levent@fbu.edu.tr

Asistan: Arş. Gör. Uğur Özbalkan

Ofis: 311

Email: ugur.ozbalkan@fbu.edu.tr

Ders Planı

- Nesneye Yönelimli Programlama (C++ dili ile)
 - Giriş
 - Sınıf ve Objelere Giriş
 - Diziler, Vektörler, İşaretçiler
 - Sınıflar
 - Operatör Aşırı Yükleme
 - Kalıtım (Inheritance)
 - Çok Biçimlilik (Polymorphism)
 - Şablonlar (Templates)
 - İstisna İdaresi (Exception Handling)
 - Dosya Okuma ve Yazma ve String İşleme
 - STL (Standard Template Library)

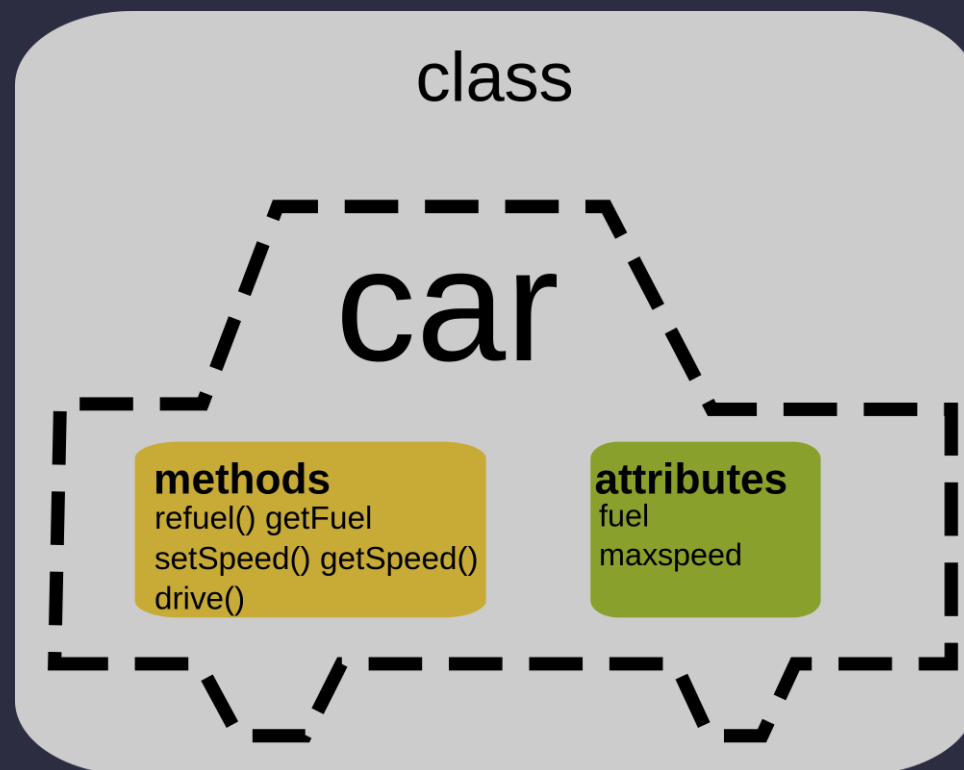
Ders Planı

- Giriş
 - C++ tarihi ve temel C++ uygulamaları



Ders Planı

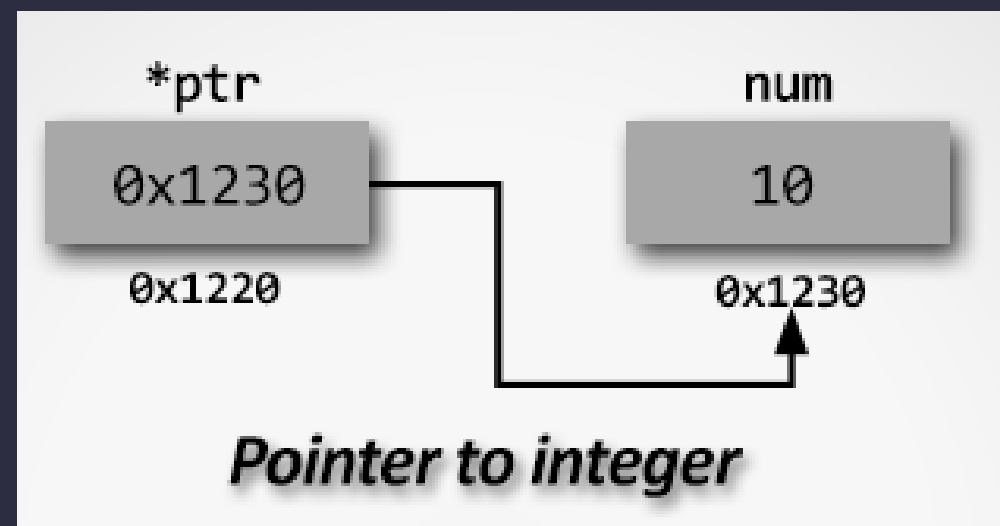
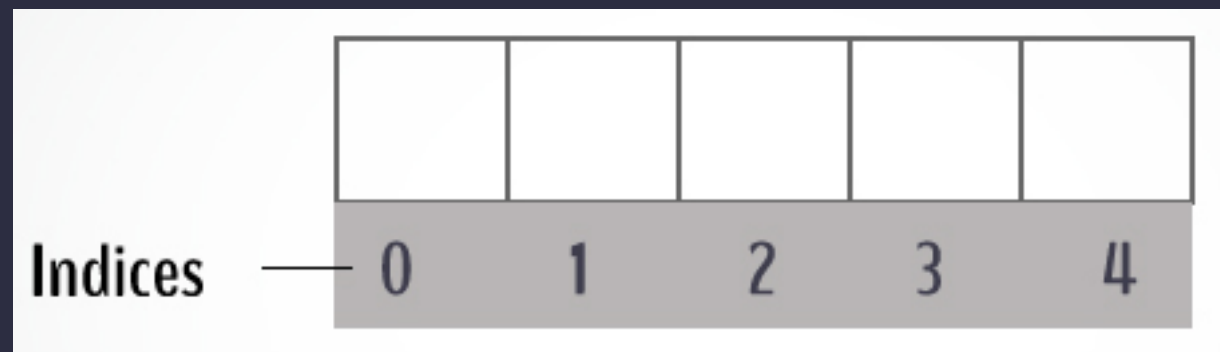
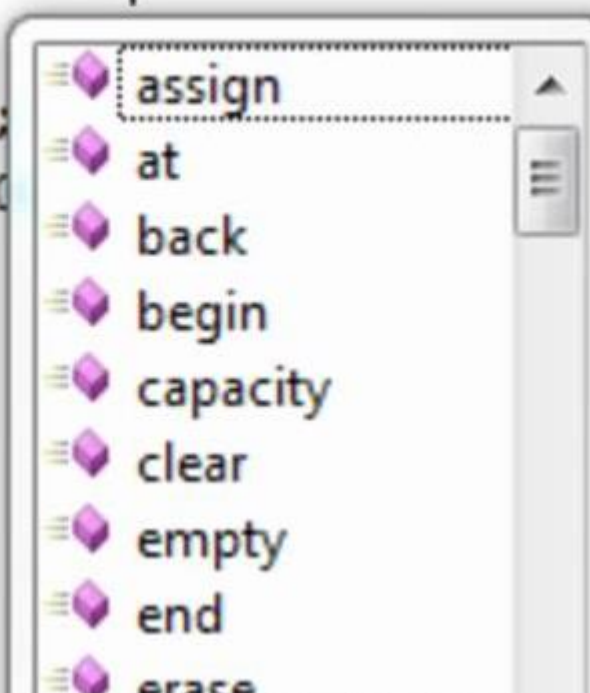
- Sınıf ve Objelere Giriş
 - Tekrar kullanılabilir kod yazılması için çeşitli yaklaşımlar



Ders Planı

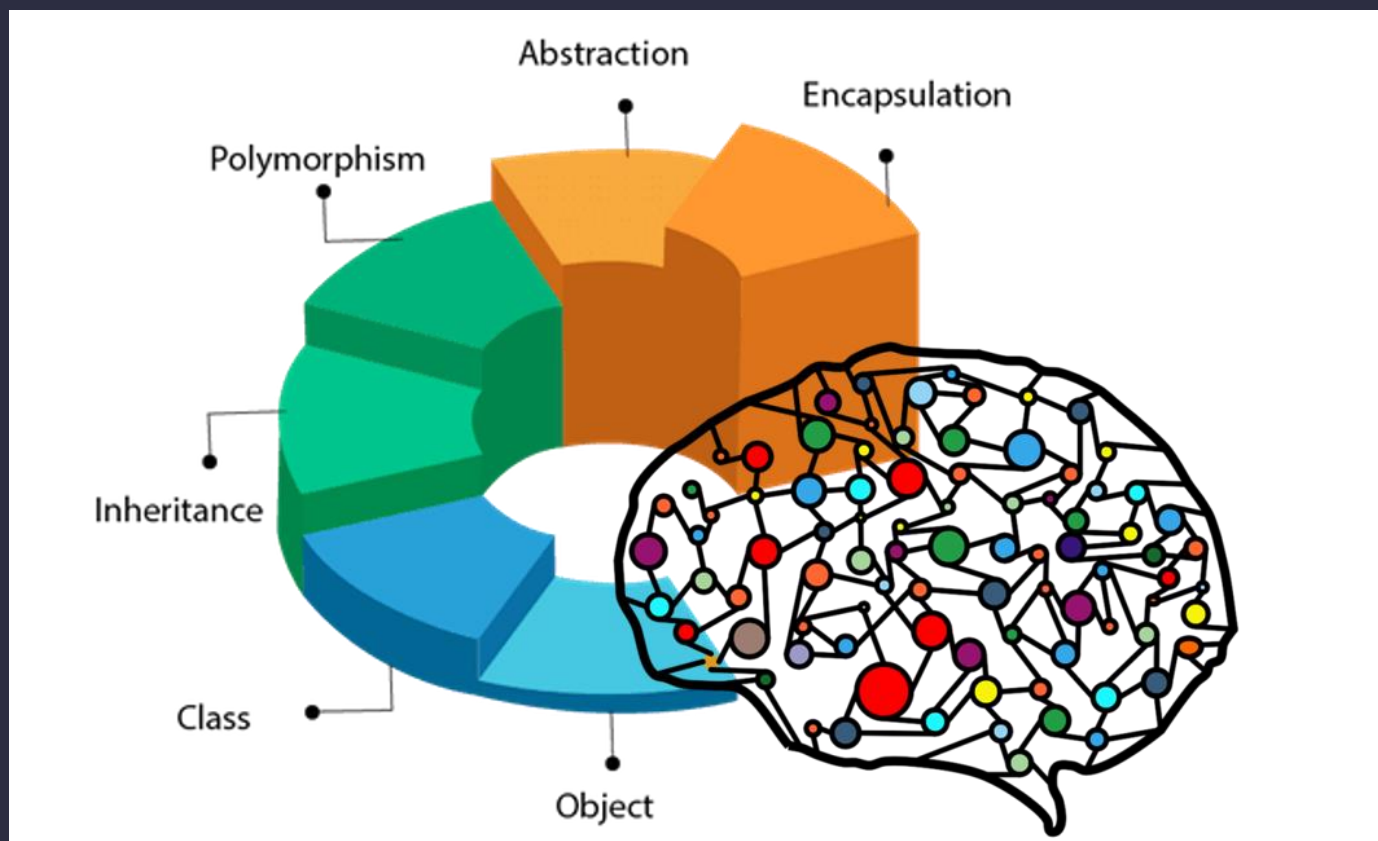
- Diziler, Vektörler, İşaretçiler

```
vector<int> VectorName;  
VectorName.  
  
getch ();  
return 0;  
}
```



Ders Planı

- Sınıflar



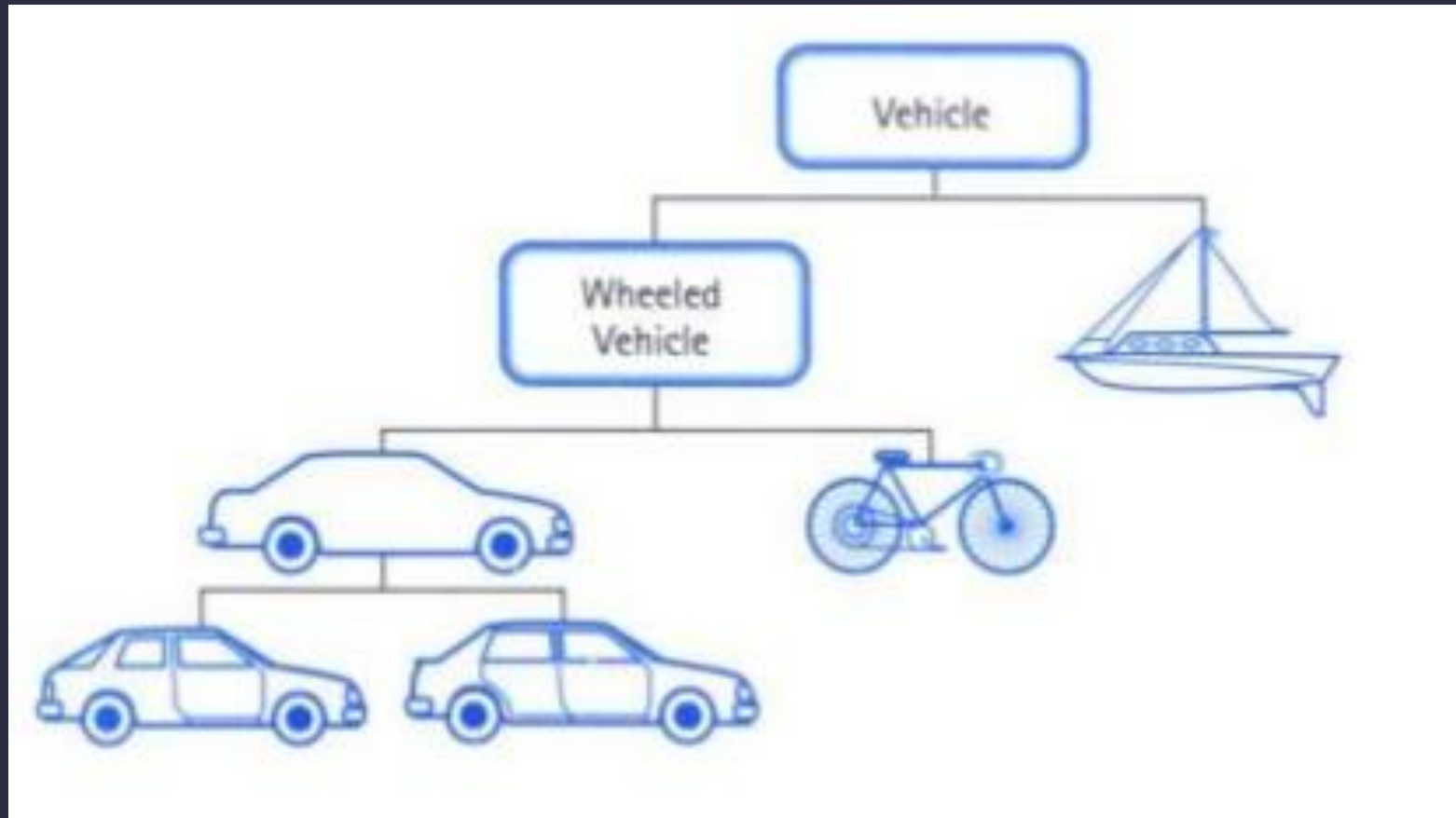
Ders Planı

- Operatör Aşırı Yükleme

```
ClassName operator - (ClassName c2) ←  
{  
    ... ..  
    return result;  
}  
  
int main()  
{  
    ClassName c1, c2, result;  
    ... ..  
    result = c1-c2;  
    ... ..  
}
```

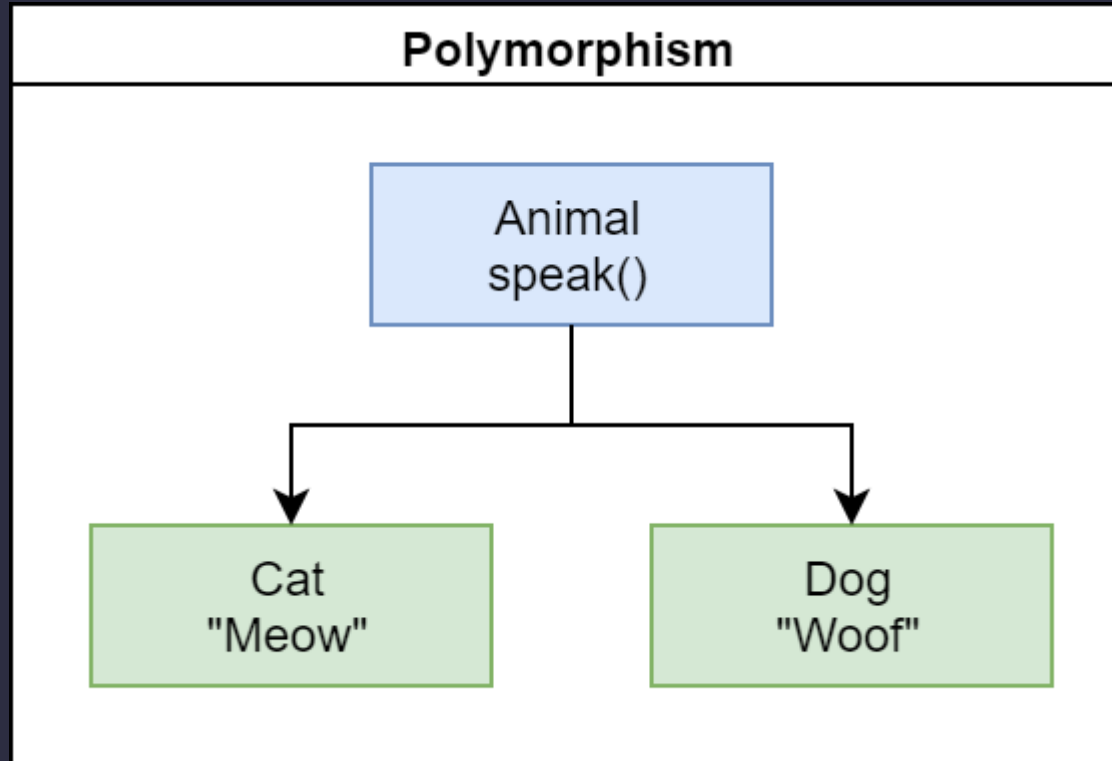

Ders Planı

- Kalıtım (Inheritance)



Ders Planı

- Çok Biçimlilik (Polymorphism)



Ders Planı

- Şablonlar (Templates)

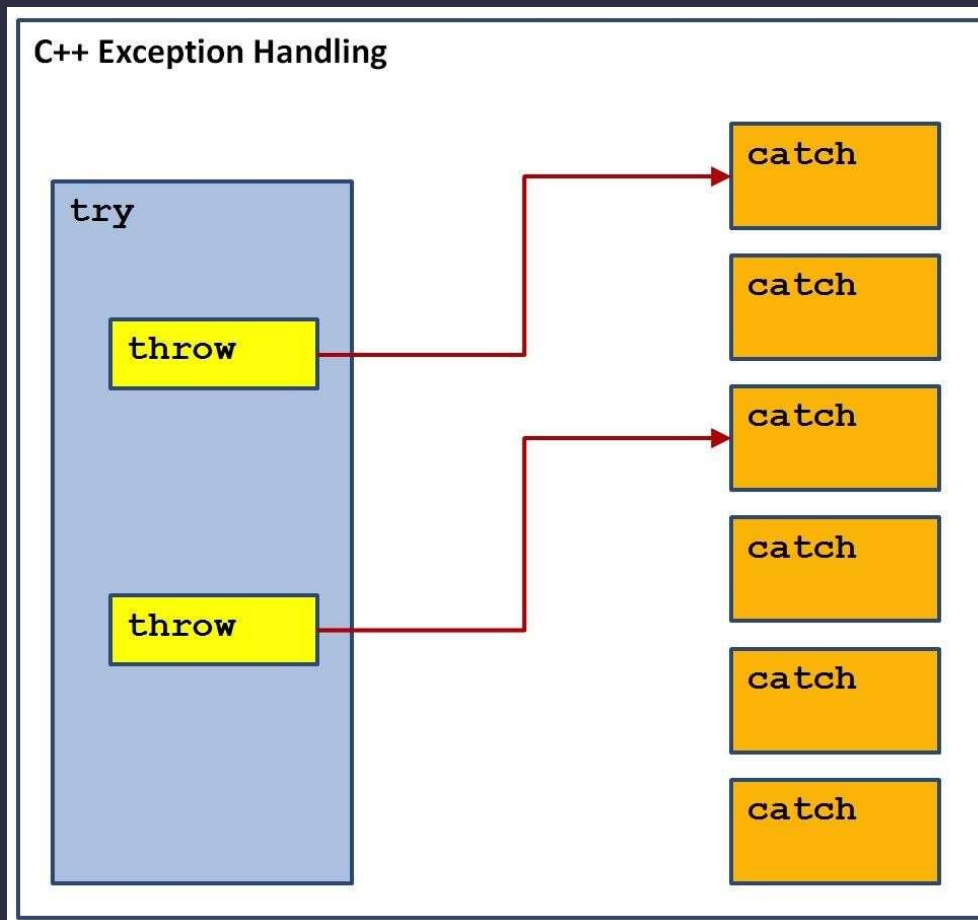
```
template <typename T>  
T min (T a, T b)  
{  
    return a < b? a: b;  
}  
  
int main()  
{  
    int x;  
    double y;  
  
    x = min<int>(10,20);  
    y = min<double>(10.2,20.9);  
}
```

←
Template
parameter must
be a type

↑
Instances of template
function

Ders Planı

- İstisna İdaresi (Exception Handling)



Ders Planı

- Dosya Okuma ve Yazma ve String İşleme

```
string String = "BillAndBob";  
String.  
get  
ret  
-}  
append  
assign  
at  
begin  
c_str  
capacity  
clear  
compare
```

Ders Planı

- STL (Standard Template Library)

```
14 {  
15     public:  
16         ContactList();  
17         void addToHead(const std::string&);  
18         void printList();  
19         void insert(const std::string&);  
20         void deleteNode(const std::string&);  
21  
22  
23     private:  
24         void reversePrintList(Contact*);  
25         Contact* head;  
26         int size;  
27  
28 }
```

STL
In
C++ language

Ders Kaynakları

Web sitesi: levent.tc

Dersler > BLM102 – Algoritmalar ve Programlama II

Ders Kaynakları

Ders Sayfası İçeriği;

- Ders Özeti
- Ders Takvimi
- Ders Notları
- Ödevler
- Projeler
- Sınavlar
- Test Sistemi
- LMS ve Piazza
- Notlar
- Geri Bildirim

Ders Kaynakları

Ders Özeti;

Ders Saatleri;

Perşembe 9.00-13.00

Ofis Saatleri;

- Dr. Öğr. Üyesi Vecdi Emre Levent - Perşembe 15.00-17.00
- Arş. Gör. Uğur Özbalkan - Salı 16.00-17.00, Cuma 16.00-17.00

Ders Kaynakları

Ders Özeti;

4-6 arasında ödev verilecektir.

2 Quiz yapılacak.

Derslere devam **%80** oranında zorunludur.

Ders Kaynakları

Ders Özeti;

Değerlendirme ağırlıkları

Ödev ve quizlerin teslim saati
üzerinden her geçen saat için
5 puan kesilecektir.

Faaliyetler	Oranlar
Vize	%20
Ödev/Quiz	%10
Lab	%15
Proje	%30
Final	%25
Bonus	5 Puan'a kadar

Ders Kaynakları

Ders Özeti;

Harf notu aralıkları

Dönem Notu	Ağırlık	Harf Notu
90-100	4.00	AA
85-89	3.50	BA
80-84	3.00	BB
75-79	2.50	CB
65-74	2.00	CC
50-64	1.50	DC
45-49	1.00	DD
0 -44	0	FF

Ders Kaynakları

Ders Özeti;

Beklenen efor

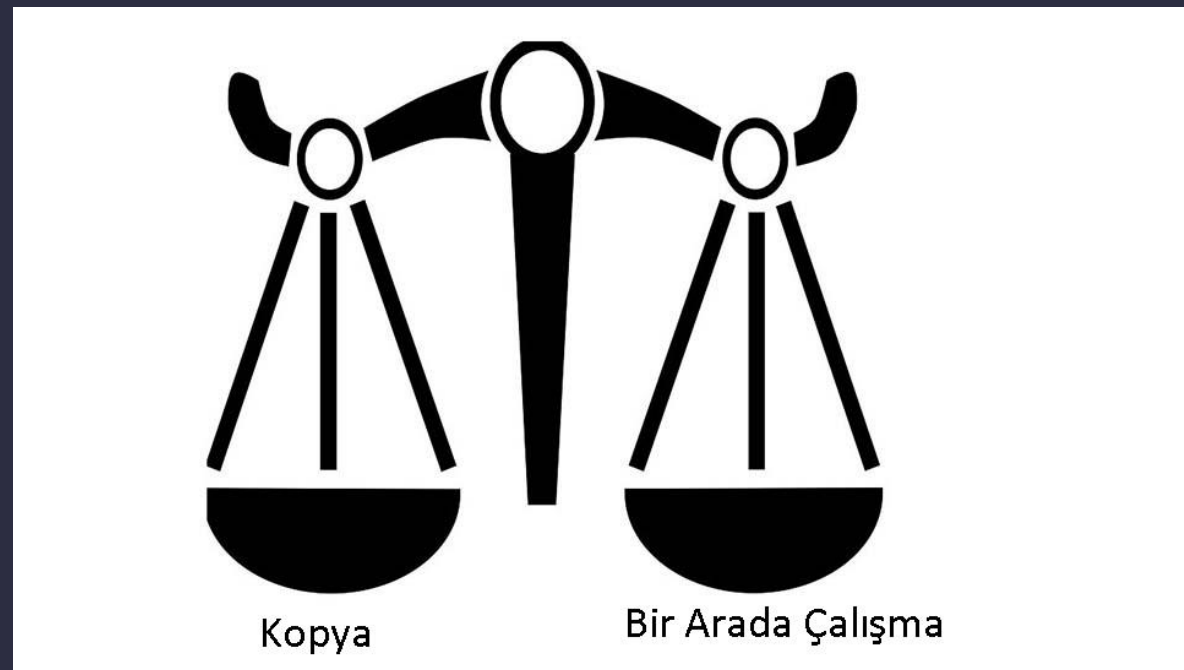
Toplam 190 saatlik
efor beklenmektedir.

İçerik	Saat	Kaç Defa	Ara Toplam
Derse Hazırlık	2	14	28
Ders Tekrarı	2	14	28
Ödev	4	6	24
Proje	48	1	48
Sınıf Dersi	4	14	56
Ara Sınav ve Final	3	2	6

Ders Kaynakları

Ders Özeti;

Akademik dürüstlük



Ders Kaynakları

Ders Takvimi;

Hafta	Konu	Tarih
1	Giriş	6.2.2020
2	Sınıflar ve Objelere Giriş	13.2.2020
3	Diziler, Vektörler ve İşaretçiler	20.2.2020
4	Sınıflar I (Classes)	27.2.2020
5	Sınıflar II (Classes)	5.3.2020
6	Operatör Aşırı Yükleme	12.3.2020
7	Kalıtım (Inheritance)	19.3.2020
8	Vize	26.3.2020
9	Çok Biçimlilik (Polymorphism)	2.4.2020
10	Şablonlar (Templates)	9.4.2020
11	Stream IO ve İstisna İdaresi (Exception Handling)	16.4.2020
12	Resmi Tatil	23.4.2020
13	Dosya Okuma/Yazma ve String İşleme	30.4.2020
14	STL (Standard Template Library)	7.5.2020
15	Final ve Proje Sunumları	Sınav Haftası

Ders Kaynakları

Ödevler;

Verilecek ödevler ve çözümleri ödev sayfasından paylaşılacaktır.

Ders Kaynakları

Projeler;

Dönem sonunda her öğrencinin yapması gereken projeler duyurulacaktır.

Ders Kaynakları

Sınavlar;

Vize ve Final sınavları için örnek soru ve sınavların çözümleri paylaşılacaktır.

Ders Kaynakları

Test Sistemi;

Çeşitli algoritma sorularının bulunduğu sistemde, problemleri çözüp algoritma geliştirme becerinizi ilerletebilirsiniz.

Ders Kaynakları

LMS ve Piazza;

LMS sistemi, size verilecek bazı ödevlerin yüklenmesini isteyeceğimiz sistemdir. Sistem otomatik olarak son yükleme tarihinde kapanacaktır.

Piazza sistemi, sınıf içi soru cevap platformudur. Ders, ödev veya sınavlar hakkında takıldığınız bir konu olduğunda, bu platforma yazabilirsiniz. Yazdığınız sorular hocalar ve öğrenciler tarafından görülür. Birbirinize de bu platform üzerinden ölçülü bir şekilde yardımcı olabilirsiniz.

Ders Kaynakları

Notlar;

Notlar sayfasında derste topladığınız tüm notlar verilmektedir.

Sayfalar arasında gezinerek vize, ödev, quiz, lab, final ve bonus'lardan kaç puan topladığınızı görebilirsiniz.

Ders Kaynakları

Geri bildirim;

Geri bildirim dersin kalitesinin artması için çok önemlidir.

Her hafta LMS sistemi üzerinden açılacak olan geribildirim mekanizmasına yorum yapabilirsiniz.

Her yorum yaptığınızda sene sonu puanınıza ek 0.5 bonus puan verilecektir.

En fazla 5 puan'lık bonus toplayabilirsiniz.

C++ Dilinin Tarihi

- İlk olarak Amerika'da Bell Laboratuvar'ında C dilinine bir eklenti olarak ortaya çıkmıştır.
- 1998 yılında ANSI (American National Standard Institute) ve International Organization for Standardization (ISO) tarafından standartlaştırılmıştır bir dildir (C++ ISO/IEC 14882:1998).
- C dilinden ayıran en büyük özelliği nesneye yönelimli bir dil olmasıdır.

C++

- Ders kapsamında nesneye yönelimli dillerin en önemli özelliği olan sınıf kavramı ve C++ kütüphaneleri ile çalışılacaktır.
- Sınıf kavramı ile tasarlanmış olan bir kod, başka bir programcı tarafından kullanımı, sınıf yapısı ile tasarlanmamış olan bir koda göre daha kolaydır.

- Dolayısıyla sınıf kavramı, geliştirilen kodların başka projelerde yeniden kullanımını kolaylaştırır.

C++

```
#include <iostream> // allows program to output data to the screen

// function main begins program execution
int main()
{
    std::cout << "Welcome to C++!\n"; // display message

    return 0; // indicate that program ended successfully
} // end function main
```

C++

```
#include <iostream> // allows program to output data to the screen

// function main begins program execution
int main()
{
    std::cout << "Welcome to C++!\n"; // display message

    return 0; // indicate that program ended successfully
} // end function main
```

- cout ifadesi, "Standart çıkış akışı objesi (standard output stream object)" dir.
- cout objesine << ifadesi ile, sağ tarafındaki içerik yazılmış olur.
- cout objesi ise genellikle monitöre bağlıdır. Dolayısıyla ekrana çıktı verilir.

C++

```
#include <iostream> // allows program to output data to the screen

// function main begins program execution
int main()
{
    std::cout << "Welcome to C++!\n"; // display message

    return 0; // indicate that program ended successfully
} // end function main
```

- cout'un başındaki std:: ifadesi ise bir önışlem talimatıdır (preprocessor directive).
- std:: ifadesi ile, std isim uzayındaki (namespace) cout objesinin kullanıldığı ifade edilmiştir.

C++

```
#include <iostream>
using namespace std;

// first name space
namespace first_space {
void func() {
cout << "Inside first_space" << endl;
}
}

// second name space
namespace second_space {
void func() {
cout << "Inside second_space" << endl;
}
}
```

```
int main() {
// Calls function from first name space.
first_space::func();

// Calls function from second name space.
second_space::func();

return 0;
}
```

- Namespace kavramı, kütüphanelerdeki fonksiyon ve değişken isimlerinin aynı olabileceği için, bir tanımlama daha yapılarak karışıklığı önlemektedir.

C++

```
#include <iostream> // allows program to perform input and output

// function main begins program execution
int main()
{
    // variable declarations
    int number1; // first integer to add
    int number2; // second integer to add
    int sum; // sum of number1 and number2

    std::cout << "Enter first integer: "; // prompt user for data
    std::cin >> number1; // read first integer from user into number1

    std::cout << "Enter second integer: "; // prompt user for data
    std::cin >> number2; // read second integer from user into number2

    sum = number1 + number2; // add the numbers; store result in sum

    std::cout << "Sum is " << sum << std::endl; // display sum; end line
} // end function main
```

```
#include <iostream> // allows program to perform input and output

using std::cout; // program uses cout
using std::cin; // program uses cin
using std::endl; // program uses endl

// function main begins program execution
int main()
{
    int number1; // first integer to compare
    int number2; // second integer to compare
```



```
cout << "Enter two integers to compare: "; // prompt user for data
cin >> number1 >> number2; // read two integers from user

if ( number1 == number2 )
    cout << number1 << " == " << number2 << endl;

if ( number1 != number2 )
    cout << number1 << " != " << number2 << endl;

if ( number1 < number2 )
    cout << number1 << " < " << number2 << endl;

if ( number1 > number2 )
    cout << number1 << " > " << number2 << endl;

if ( number1 <= number2 )
    cout << number1 << " <= " << number2 << endl;

if ( number1 >= number2 )
    cout << number1 << " >= " << number2 << endl;
} // end function main
```

C++

Tüm std elemanlarını kullanmak için;

- `using namespace std;`

yazılır.