

# Table of Contents

<b>C++</b> .....	3
<b>Başlangıç ve Kurulum</b> .....	3
<b>Fonksiyonlar ve Çoklu Dosya Geliştirme</b> .....	3
<b>Hata Ayıklama (Debugging) ve Hata Yönetimi</b> .....	3
<b>Temel Veri Tipleri ve Sabitler</b> .....	3
<b>Operatörler ve Bit İşlemleri</b> .....	4
<b>Kapsam, Yaşam Süresi ve Bağlantı</b> .....	4
<b>Akış Kontrolü</b> .....	4
<b>Tip Dönüşümleri ve Çıkarımları</b> .....	4
<b>İşaretçiler, Referanslar ve Dinamik Bellek</b> .....	4
<b>Enum'lar ve Struct'lar</b> .....	4
<b>Diziler, Vektörler ve Algoritmalar</b> .....	4
<b>İleri Düzey Fonksiyonlar ve Şablonlar</b> .....	5
<b>Nesne Yönelimli Programlama: Sınıflar</b> .....	5
<b>Operatör Aşırı Yükleme</b> .....	5
<b>Nesne İlişkileri</b> .....	5
<b>Kalıtım ve Sanal Fonksiyonlar</b> .....	5
<b>İleri Düzey Şablonlar</b> .....	5
<b>İstisnalar</b> .....	5
<b>Taşıma Semantiği ve Akıllı İşaretçiler</b> .....	6
<b>Girdi ve Çıktı (I/O) Akışları</b> .....	6



# C++

Snippet from [Wikipedia](#): **C++**

**C++** (/ˈsiː plʌs plʌs/, telaffuz: *si pılas pılas*), Bjarne Stroustrup tarafından 1979 yılında Bell Laboratuvarları'nda geliştirilmeye başlanmış, C'yi kapsayan ve çok paradigmatlı, yaygın olarak kullanılan, genel amaçlı bir programlama dilidir.

İlk olarak *C With Classes* (Sınıflarla C) olarak adlandırılmış, 1983 yılında ismi C++ olarak değiştirilmiştir. Günümüzde en çok kullanılan programlama dillerinden biri olmuştur.

C++ tasarlanırken C programlama dili ile olabildiğince uyumlu olması göz önüne bulundurulmuş ve K&R2'deki tüm örnek kodun derleneceği şekilde tasarlanmıştır.

C++, C'nin sağladığı alt seviye sıkı donanım desteğinin yanında farklı veri türleri, sınıf, template, sıradışı durum yönetimi, isim alanı (namespace), işleç fazladan yüklemesi, işlev fazladan yüklemesi, referans, hafıza yönetimi ve pek çok kütüphane imkanı sunar.

[Creative Commons Attribution-Share Alike 4.0](#)

## Başlangıç ve Kurulum

- [Derleyici ve Bağlayıcı \(Compiler & Linker\)](#)
- [Geliştirme Ortamı \(IDE\)](#)
- [İlk Program](#)
- [C++ Temelleri](#)

## Fonksiyonlar ve Çoklu Dosya Geliştirme

- [Fonksiyon Temelleri](#)
- [Ön Bildirimler \(Forward Declarations\)](#)
- [Çoklu Dosya Programları](#)
- [Önişlemci \(Preprocessor\)](#)
- [İsim Alanları \(Namespaces\)](#)

## Hata Ayıklama (Debugging) ve Hata Yönetimi

- [Hata Ayıklama Taktikleri](#)
- [Debugger Kullanımı](#)
- [Hata Tespiti](#)
- [Kod Kapsamı \(Code Coverage\)](#)

## Temel Veri Tipleri ve Sabitler

- [Temel Tipler](#)
- [İşaretili \(Signed\) vs İşaretsiz \(Unsigned\)](#)
- [Sabitler ve Değişmezler](#)

- [Modern Metinler \(Strings\)](#)

## Operatörler ve Bit İşlemleri

- [Operatörler](#)
- [Öncelik ve Birleşme Yönü](#)
- [Bit Manipülasyonu \(Opsiyonel\)](#)
- [Bitsel \(Bitwise\) Operatörler](#)

## Kapsam, Yaşam Süresi ve Bağlantı

- [Kapsam \(Scope\) ve Bloklar](#)
- [Bağlantı \(Linkage\)](#)
- [İsim Alanı Derinlikleri](#)
- [Satır İçi Değişkenler \(Inline\)](#)

## Akış Kontrolü

- [Koşullar](#)
- [Switch İfadeleri](#)
- [Döngüler](#)
- [Rastgele Sayılar](#)

## Tip Dönüşümleri ve Çıkarımları

- [Kapalı ve Açık Dönüşümler](#)
- [Tip Takma İsimleri](#)
- [Tip Çıkarımı \(Type Deduction\)](#)

## İşaretçiler, Referanslar ve Dinamik Bellek

- [L-values ve R-values](#)
- [Referanslar](#)
- [İşaretçiler \(Pointers\)](#)
- [Dinamik Tahsisat \(Geleneksel\)](#)

## Enum'lar ve Struct'lar

- [Numaralandırmalar](#)
- [Struct Yapıları](#)
- [Sınıf Şablonları ve CTAD](#)

## Diziler, Vektörler ve Algoritmalar

- [Dinamik Diziler \(std::vector\)](#)
- [Sabit Boyutlu Diziler \(std::array\)](#)
- [Eski Nesil Diziler](#)
- [İteratörler ve Algoritmalar](#)

## İleri Düzey Fonksiyonlar ve Şablonlar

- Fonksiyon Aşırı Yükleme (Overloading)
- Fonksiyon Şablonları (Templates)
- Gelişmiş İşaretçiler
- Lambda İfadeleri

## Nesne Yönelimli Programlama: Sınıflar

- Sınıf Temelleri
- Kurucular (Constructors)
- Gizli 'this' İşaretçisi
- Yıkıcılar (Destructors)
- Friend ve Static Kullanımı

## Operatör Aşırı Yükleme

- Aşırı Yükleme Yöntemleri
- Yaygın Operatörler
- İleri Düzey
- Kopyalama Mekanizmaları

## Nesne İlişkileri

- Kompozisyon (Composition)
- Agregasyon (Aggregation)
- Birliktelik ve Bağımlılık
- Konteyner Sınıflar

## Kalıtım ve Sanal Fonksiyonlar

- Kalıtım (Inheritance)
- Çoklu Kalıtım
- Polimorfizm
- Arayüzler (Interfaces)
- Dinamik Tip Dönüşümü

## İleri Düzey Şablonlar

- Sınıf Şablonları (Class Templates)
- Tipten Bağımsız Parametreler
- Şablon Özelleştirme

## İstisnalar

- Try, Catch, Throw
- İstisna Sınıfları
- İstisna Belirleyiciler

## Taşıma Semantiği ve Akıllı İşaretçiler

- [R-value Referansları \(&&\)](#)
- [Taşıma Semantiği \(Move Semantics\)](#)
- [Akıllı İşaretçiler](#)

## Girdi ve Çıktı (I/O) Akışları

- [Akışlar \(Streams\)](#)
- [Metin Akışları \(String Streams\)](#)
- [Dosya İşlemleri \(File I/O\)](#)
- [Durum Yönetimi](#)

Bu belgedeki bilgiler [UCH Viki](#)'den alıntılanmıştır.

From:

<https://wiki.ulascemh.com/> - **UCH**

Permanent link:

<https://wiki.ulascemh.com/doku.php?id=tr:cs:lang:cpp:start>

Last update: **2026/04/02 16:13**

