

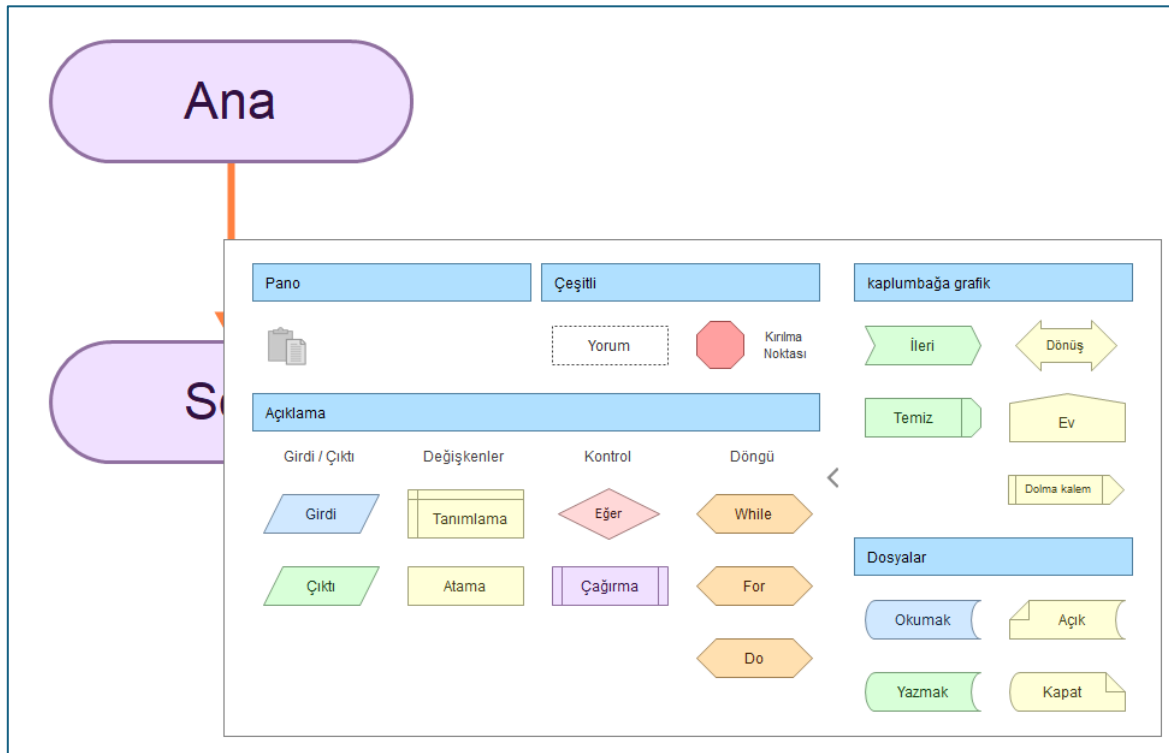
Bölüm 1. Programlamaya Giriş ve Anaconda Navigator Kurulumu

Algoritma

Herhangi bir sorunun çözümü için izlenecek yolun tarifi algoritma olarak adlandırılır. Çözüme ulaşmak amacıyla yapılması gereken aşamalar adım adım ve net şekilde belirtilir. Algoritma **Başla** ile başlatılır ve **Bitir** ile bitirilir.

Akış diyagramı

Algoritmanın görsel olarak simge ya da sembollerle ifade edilmesidir. Akış diyagramı hazırlamak için [flowgrothim](https://flowgrothim.com) isimli ücretsiz uygulama kullanılabilir.



Şekil 1.1. **Flowgrothim** uygulamasının ekran görüntüsü

Her algoritma Başla ile başlanıp Bitir ile bitirildiğinden akış diyagramı da bu ifadelerle başlayıp biter. Flowgrothim uygulamasında Başla ve Bitir yerine **Ana** ve **Son** ifadeleri kullanılmıştır, bu ifadeler Başla ve Bitir ile aynı anlamdadır. Programın Türkçe dil desteği bulunmaktadır ancak bazı programlama dilleri Türkçe dilini desteklemediğinden değişken tanımlarında Türkçe karakter kullanılmaması tercih edilebilir.

Bölüm 1. Programlamaya Giriş ve Anaconda Navigator Kurulumu

Örnek: İki sayının toplamını bulan programın algoritmasını ve akış diyagramını hazırlayınız.

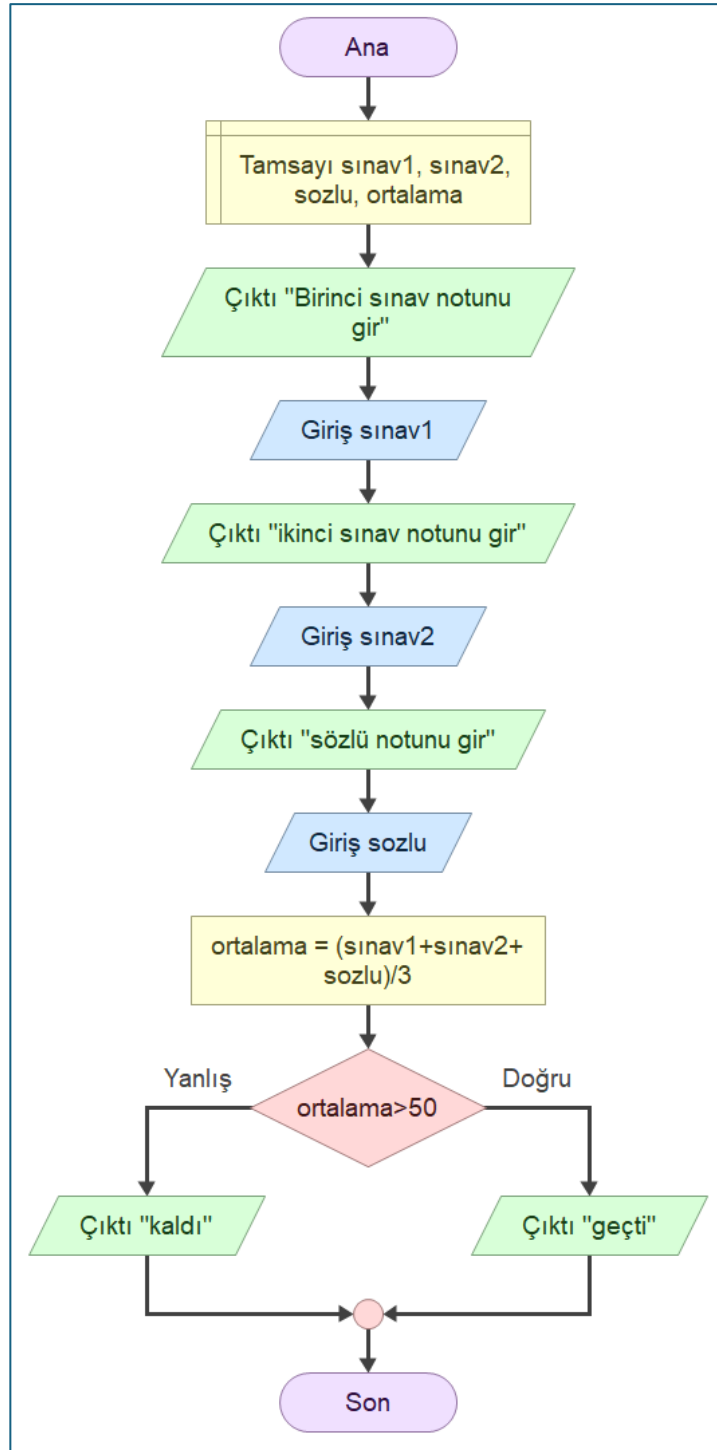
1. Başla
2. Birinci sayıyı giriniz
3. İkinci sayıyı giriniz
4. 1. ve 2. sayının toplamını hesapla
5. Sonucu yazdır
6. Bitir



Bölüm 1. Programlamaya Giriş ve Anaconda Navigator Kurulumu

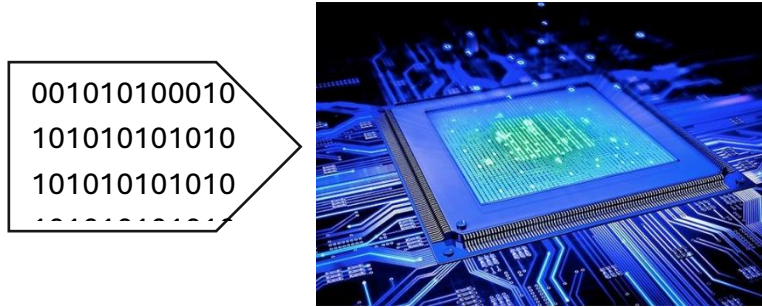
Örnek: 2 adet yazılı 1 adet sözlü sınavın ortalamasını hesaplayarak ortalama değer 50 değerinden büyükse ekrana “geçer” yazdıran küçükse “kalır” yazan programın algoritmasını hazırlayınız.

1. Başla
2. Birinci sınav notunu gir
3. İkinci sınav notunu gir
4. Sözlü notunu gir
5. Ortalamayı hesapla
6. Eğer ortalama 50’den büyükse 7. adıma git değilse 8. adıma git
7. Ekrana “geçti” yaz
8. Ekrana “kaldı” yaz
9. Bitir

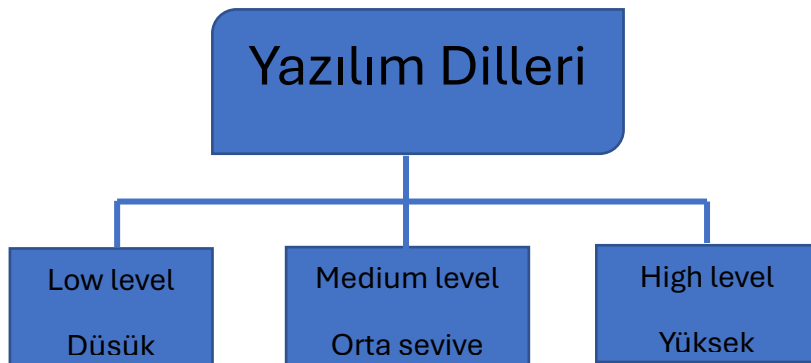


Yazılım Dillerine Genel Bakış

Mikroişlemcisi olan cep telefonu, bilgisayar gibi aygıtlar sadece 0 ve 1' ler ile yani Makine dili ile çalışır. Mikroişlemcinin bu komutları anlayabilmesi için komutlar kodlanmalıdır.



Yazılım dilleri, makine dili kodlarını kolaylık olsun diye semboller ve yazılım diline özgü gramer ile ifade etme aracıdır. Bu sembol ve gramerin, mikroişlemcinin anlayacağı 0 ve 1' lere dönüştürülmesi gerekir. Bu dönüştürme işlemine **derleme** adı verilir.



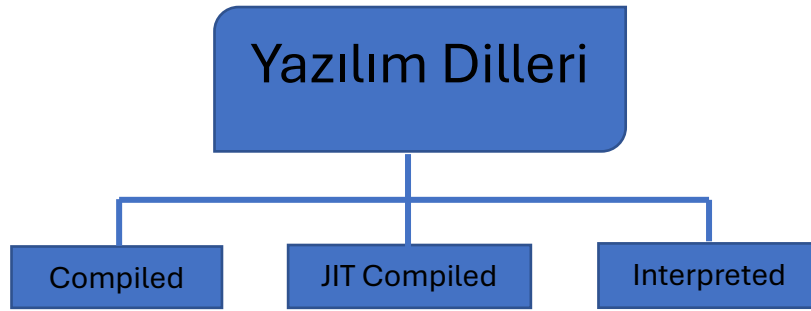
Düşük seviye yazılım dillerinde, makine dili sembollerle ifade edilir ve Assembly dili olarak adlandırılır. Bu dilde yazılan kodlar uzundur ve derlendikten sonra daha kısa bir kod bütününe dönüşür. Assembly dili etkin bir dildir fakat insan dil yapısına benzemediğinden hem bu dili öğrenmek hem de bu dilde kodlama yapmak zordur. Bu nedenle de Orta seviye yazılım dilleri oluşturulmuştur.

Bölüm 1. Programlamaya Giriş ve Anaconda Navigator Kurulumu

Orta seviye yazılım dilleri, insan dil yapısına daha yakın olup, düşük seviye dilinin de özelliklerini bünyesinde barındırır. Bu nedenle hem insan diline hem de makine diline yakındır. Orta seviye dillerin derlenmesi ile makine diline dönüştürülen kodlar, düşük seviye diller kadar etkili olmasa da yine de performansları iyidir. Orta seviye dillere örnek olarak C dili veya C++ verilebilir.

Yüksek seviye yazılım dilleri, insan diline yakındır. Bu nedenle öğrenmesi ve kodlama süreci çok kısadır. Bu yazılım dillerinde çok sayıda yazılım kütüphanesi bulunur. Bu kütüphaneler sayesinde yazılımcı uzun uzun kodlar kullanmadan bu kütüphanelerden faydalanır. Dahası farklı kütüphanelerdeki özellikleri amaca uygun olarak farklı kombinasyonlarda kullanarak etkin kodlar oluşturulabilir. Örnek olarak **C#**, **Python** ve **Java** verilebilir.

Yazılım dilleri, çalışmaları açısından **Compiled** (derlenebilen), **JIT-Just in Time Compiled** (Zamanında derlenen) ve **Interpreted** (Yorumlanan) yazılım dilleri olarak da üçe ayrılır.



Derlenen dillere örnek olarak **C**, **C++** ve **Pascal** verilebilir.

Zamanında derlenen yazılım dillerinde bir veya daha fazla sanal makine vardır. Bu dillere örnek olarak **C#** ve **Java** verilebilir.

Yorumlanan yazılım dillerinde, yazılımcının yazdığı kod, satır satır okunup, yorumlanıp çalıştırılır. Bu nedenle bu dil grubu, bir yorumlayıcıya ihtiyaç duyar. Yorumlanan dillere en iyi örnek **Python**’ dur.



NOT: “En iyi yazılım dili” diye bir durum yoktur. Her yazılım dili yerinde kullanıldığı zaman en iyi yazılımdır.

Python Programlama Dili

Python (*paytın* olarak okunur), nesne yönelimli, yorumlamalı, modüler ve etkileşimli yüksek seviye programlama dilidir. Diğer programlama dillerinden en önemli farkı sözdiziminin basit ve bu nedenle de akılda kalıcı ve daha kısa kodlar ile aynı işlemin yapılabilir olmasıdır.

Geliştirilmeye 1990 yılında **Guido van Rossum** tarafından Amsterdam'da başlanmıştır. Adını sanılanın aksine bir yılından değil Guido van Rossum'un çok sevdiği, **Monty Python** adlı altı kişilik bir İngiliz komedi grubunun **Monty Python's Flying Circus** adlı gösterisinden almıştır. Günümüzde Python Yazılım Vakfı çevresinde toplanan gönüllülerin çabalarıyla sürdürülmektedir. Python 1.0 sürümüne Ocak 1994'te ulaşmıştır. Son kararlı sürümü, 2.x serisinde Python 2.7 ve 3.x serisinde Python 3.5.2'dir. 3 Aralık 2008 tarihinden itibaren 3.x serisi yayınlanmaya başlamıştır; ancak 3.x serisi ile 2.x serisi uyumlu değildir.

- İlgilenen öğrenciler için ek Türkçe kaynak: [Python Programlama Dili- Python 3 Türkçe Kaynak](#)
- [Python resmî sitesi](#)

Python Programlama Dili

Python dili, öğrenilmesi kolay ve güçlü bir programlama dili olması yanında;

- Kolay ve net okunabilen yazımı
- Dinamik veri yapısı
- Güçlü ifade yeteneği
- Modüler yapısı ve geniş kütüphanesi
- Nesne tabanlı programlamaya uygun olması
- Exeption tabanlı hata yönetimi
- C ve C++ ile ek modüller yazma kolaylığı
- Diğer programlara kodlama arayüzü olarak dahil edilebilmesi

özelliklerine sahiptir. Python dili;

- Flask ve Django framework' leri ile web geliştirme
- Ağ ve socket programcılığı
- Arayüz geliştirme
- Requests, Beautiful Soup ya da Scrapy ile örümcek türü yazılımlar geliştirmede
- Scikit-learn ve Numpy modülleri ile Machine Learning (Makine öğrenmesi) ve Yapay Zeka çalışmalarında kullanılır.

Bölüm 1. Programlamaya Giriş ve Anaconda Navigator Kurulumu

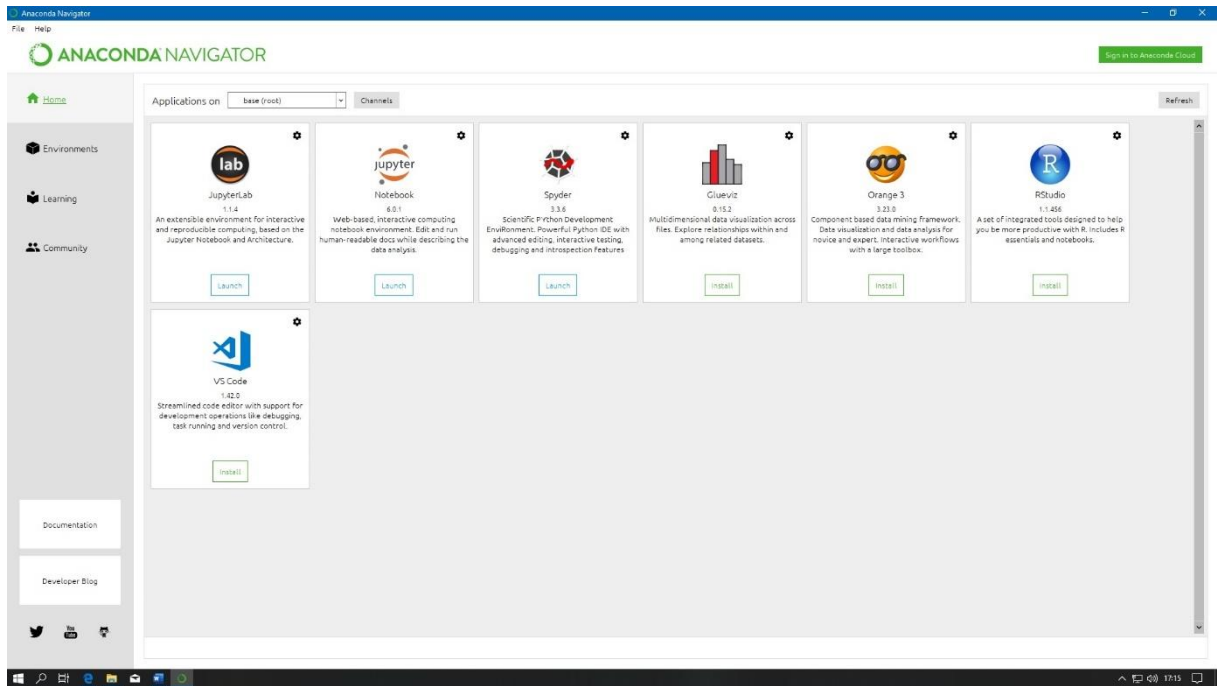
Python için Gerekli Ortamların Kurulması

Anaconda (<https://www.anaconda.com/distribution/>) adresinden **Anaconda Distribution** programı indirilip kurulmalıdır. Anaconda dağıtımının özelliği, bilimsel uygulamalar ve makine öğrenmesi için gerekli olan numpy, pandas, matplotlib gibi kütüphaneleri içermesidir.

Windows için **Ayarlar → Sistem → Hakkında** kısmına girerek ya da masaüstünde bulunan **Bu Bilgisayar'** a sağ tıklayıp, özellikler seçilerek bilgisayarınızın bit sayısını öğrenebilirsiniz. Bilgisayarınızın bit sayısına göre 64-bit ya da 32-bit olarak indirilmelidir.

Anaconda yüklenirken **Next, I Agree, Just me** ya da **All users** (bu seçim kullanıcıya aittir) seçenekleri ile Anaconda sisteme yüklenir. Bu kurulum ile **Jupyter Notebook, Phyton ara yüzü** ve **Python yorumlayıcısı** bilgisayarınıza yüklenmiş olacaktır.

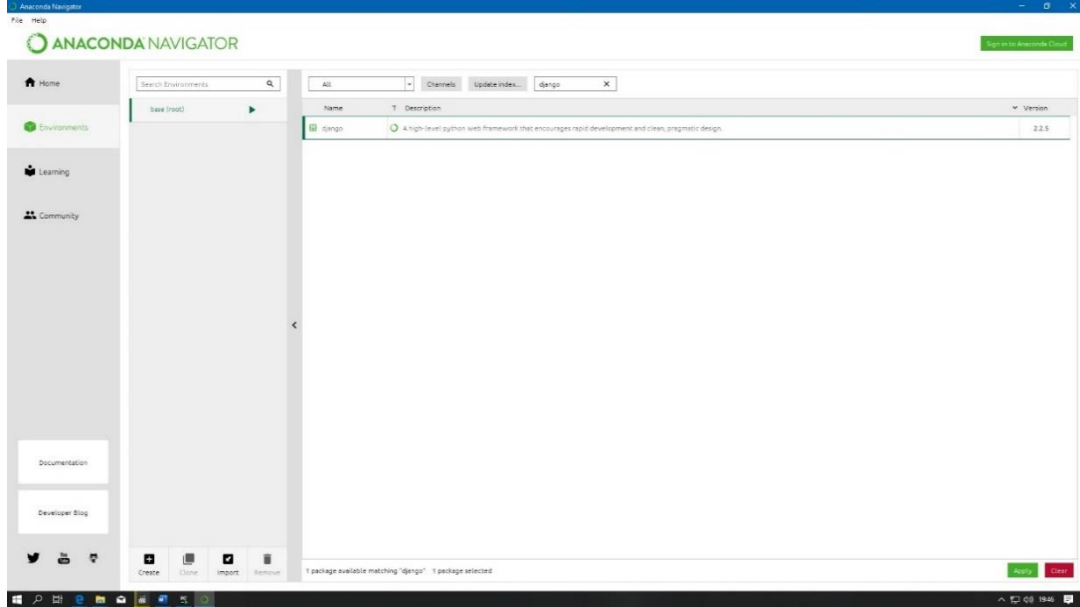
Windows' un arama bölümüne Anaconda Navigator yazılarak Anaconda başlatılır.



Anaconda açıldığında, Anaconda ile birlikte kurulan paketlerin görüntülediği yukarıdaki pencere açılır. Anaconda'yı açmanın bir diğer yolu da Anaconda'nın kurulduğu klasöre giderek Scripts klasöründen **Anaconda Navigator.exe** dosyasını çift tıklamaktır.

Bölüm 1. Programlamaya Giriş ve Anaconda Navigator Kurulumu

Anaconda Navigator'ün Environments sekmesine girilirse, Anaconda ile birlikte gelen Python paketleri görüntülenebilir. Bu bölümde daha sonra kullanılacak birçok uygulama halihazırda yüklenmiş olur. Ancak Django yüklü değildir. Django'yu yüklemek için, Anaconda Navigator'de Environments'a girilir. Sağ taraftaki bölümde **All** seçilerek arama (search) bölümüne Django yazılarak arama yapılır. Django'nun yüklü olmadığı görülür. Yüklemek için tike yazılarak sağ alt bölümde yer alan Apply (uygula) butonuna basılır ve Django'nun yüklenerek environments'a eklenmesi sağlanır.



Jupyter Notebook'u çalıştırmak için Anaconda Navigator Home penceresi açıkken Jupyter Notebook üzerindeki Launch butonuna basılarak Jupyter Notebook açılmış olur.

Jupyter Notebook hem çalışma esnasında hatırlatma amaçlı not alınabilen hem de Python kodlarımızı yazacağımız bir notebook olup varsayılan web ara yüzü ile çalışan bir notebook'tur. ***Jupyter notebook çalıştırıldığında bir de terminal açılır ve bu terminal kapatılmamalıdır!***

Jupyter notebook' u açmanın bir diğer yolu da Windows aramasında **cmd** yazılarak açılan ms-dos terminalinde **Jupyter Notebook** yazmaktır.