Agenda

* Algoritma nedir
* Yazılım nedir
* Makine dili
* Assembly dili
* Yüksek seviye dilli
* Kütüphaneler(laybıris)
* Libraries…..
* Backend…..

Makine dili hala oyundan çıkmış değil kullanılıyor yalnız biz Python kullanınca bu program bizim için o dile çeviriyor. (İnterpretor(yorumlayıcı/Compiler)

Algoritma Nedir ?

Bir şeyi adım adım yapmak demektir.

Bardağı kaldırıp suyu içmek için hangi işlemleri yapıyoruz.

Her şeyi hangi detayda , hangi seviyede yapmak istediğinizi bir kağıda bir programa tanımladığınız zaman siz aslında bir programın çatısını kurmuş oluyorsunuz.

Verilen girdiden beklenen çıktıyı elde etmek için kurallar dizisi

TANIM : Algoritma, belirli bir problemi çözmek veya belirli bir görevi yerine getirmek için tasarlanan adımların sistemli bir şekilde sıralanmış listesidir. Bir algoritma, bir girdi alır ve bu girdi üzerinde belirli işlemler yaparak çıktı üretir.

Günlük işlerimizde de bunu düşünebiliriz. Bunların ayrıntılarını görün.

Evrensel bir dil, işlerin yapılma dili.

Bir algoritmanın karakteristiği nedir

* İyi tanımlanmış girdiler-----🡪 Bir girdinin çok iyi tanımlanması gerekir. Girdilerimizi çok detaylı bir şekilde tanımlamalıyız. (ocağı aç değil, alttaki düğmelerden soldan ikinciyi aç demeli)
* Açık ve muğlak olmayan olmalı ----🡪 Ben bir algoritma tanımladığımda içerindeki her şey net olmalı, okuyan herkes anlamalı ve içinde tanımsız şeyler olmamalı.
* Dilden bağımsız ---🡪 algoritma dilden bağımsız olmalı: Python, c, c++, java hepsi için uygun olmalı

Konuyu en iyi biçimde karşınızdakinin konuyu hiç bilmediği şekilde yapacaksınız.

* Mümkün ----🡪 algoritmanın uygulanabilir olması gerekir

1. Hdd den 1000 tane video yu yükle
2. Bu 1000 dosyayı birleştir
3. Sonucu hdd ye yazdır

Kullanıcı ihtiyaçlarını karşılama yönünden de uygulanabilir olması gerekmektedir.

* Sonluluk---faynaytnes🡪 Algoritmanızın sonu olmalı ( sokağa çık yürü dedik bitti, 1 den başla 2 şer ikişer saydır diyoruz )

1. A=1
2. Bir üstteki adıma geç
3. Başla
4. Sıcaklığı oku
5. Ekrana yazdır
6. Git 1

* İyi tanımlanmış çıktılar

girdi

Verilen girdiden beklenen çıktıyı elde etmek için belirlenen kurallar kümesi.

Çıktı

Bir algoritma ------- Çay Demleme Algoritması

Çay poşetini bir bardağa koyun. (kaç tane , nereden alacağım, hangi çayı)

Su ısıtıcısını suyla doldurun. (ne kadar dolacak)

Su ısıtıcısındaki suyu kaynatın.

Kaynamış suyun bir kısmını bardağa dökün.

Bardağa süt ekleyin.

Bardağa şeker ekleyin.

Çayı karıştırın. (sonsuz döngü olabilir)

Çayı iç.

Algoritma tanımlarken bazı standartlar var

Az önce gördüğümüz gibi yazı olarak da tanımlaya biliriz tabi numara vermeliyiz başlarına.

Ya da grafiksel olarak da algoritma tanımlayabirilirz.

Telefon et

Evde misin

Mesaj bırak

Benimle yemek yemek ister misin

Birlikte yemek

Bir şeyler içmek ister misin

Benimle dışarda bir şeyler yapmak ister misin 5 kere sor…..

Evet se ne yapalım

<https://www.google.com/search?q=exact+instructions+challenge+youtube&oq=exact+instructions+challenge&aqs=chrome.5.69i57j0i512l9.25687j0j15&sourceid=chrome&ie=UTF-8#fpstate=ive&vld=cid:22a5e4f6,vid:Ct-lOOUqmyY>

Algoritmanın farklı şekillerde yazılabileceğini söylemiştik

Pisedocod--🡪 kod gibimsi ------ Algoritma geliştirmemize yardımcı olur

Toplamı sıfıra eşitle

Rakamların listesini al

Loop---- Listedeki her bir rakam için döngü kur

Her sayıyı toplam son döngüye ekleyin ---------------------------- örnek : 45, 2, 3, 7, 11

İf eğer sayı 0 dan büyükse ….

Küçükse …….

Pseudocode

Algoritma geliştirmemizde yardımcı olan bir araç. Algoritma geliştirmek için bir methottur.

Sorun tanımlarken o sorunu önce metne dökmek lazım, sorunu kod gibi bilgisayar mantığında metine dökmektir pisedocod.

Pisedokod gayri resmi üst seviye üst dilden bir tanımlama, neyin tanımlaması;

Bir bilgisayar programının işletme çalışma prensibinin tanımlanması ya da algoritmanın da olabilir.?????????? Yani biz bu işleri bu yöntemle yapıyoruz bu yöntemleri kullanarak bir algoritma geliştir diyebilirler.

Algoritma

Bir problemin nasıl çözüleceğine ait kesin belirtim

-muğlak olmayan tanımlamalar. Bir problemin çözümünü muğlaklıktan uzaklaşmış, net algoritmanın karakterleri özelliklerini taşıyacak şekilde bir akışa dönüştürmesi.

Bir bilgisayar programının veya başka bir algoritmanın çalışma ilkesinin gayri resmi üst düzey açıklaması

Sorunu basitleştirmeye ve anlamaya yardımcı olur

-problemin anlaşılmasını sağlayan ve basitleştiren bir yardımcıdır.

Algoritma geliştirme yöntemi

YAZILIM NEDİR ? Bilgisayarın ruhu gibi düşünebiliriz

* Hardwere de olduğu gibi fiziksel olarak dokunulamaz ama etkisi görülür
* Bilgisayar donanımı ile işlenilen veri arasındaki bağdır. veriyi işler dedik saklar ve çıktı olarak verir dedik.
* Çalıştırılmadan önce bilgisayarın hafızasına yüklenmelidir.

Biz bu yazılımlı çalıştırmak için önce bunları harddisk e yüklememiz gerekiyor.

* RAM e yüklenmeden hiçbir program çalışmaz çünkü işlemci komutları sırayla nereden okuyordu, ram den okuyordu.
* Aynı zamanda yazılım bilgisayarın özel bir görevi gerçekleştirmek için sırayla çalıştırdığı önceden yazılan talimatlardır (algoritmada yazmıştık )

Önceden yazılmış komutlar bir bilgisayar kodlama diliyle yazılıyor

* Genel olarak diller C, C++, Basic, Java , Python dır.

Biz işte bu kodlama dilleriyle biz kodlarımızı yazıyoruz, bunları yüklenebilir hale getiriyoruz ve daha sonra bunları bilgisayarımıza yükleyip çalıştırabiliyoruz. Tabi bunların hepsini bilgisayarımıza yüklememiz gerekmiyor. Bazı oyun ve programlar web tabanlı olabiliyor.

Yazılım mimarilerine bakıldığında bilgisayarlarımızda kullanabileceğimiz yazılımlara baktığımızda neler vardır;

Aslında bilgisayarımız donanımdan ibaret ama üzerinde bu donanıma en yakın yazılım işletim sistemleridir, bundan da daha yakın olan bilgisayara hükmeden çekirdek, Kernel yazılımı vardır.

Ve bu işletim sisteminin üzerine çıktığımızda diğer yazılım uygulamaları vardır.

Herhangi bir programlama dili duydunuz mu ?

Low levıl lenguiç--🡪düşük seviyeli diller, daha anlaşılması zor, kullanılması zor diller

Hay levıl lenguiç --🡪yüksek seviyeli diller, Kullanıcıların daha kolay program yazdığı diller olarak bahsedebiliriz

Yüksek seviye diller yorumlayıcılardan geçerler

Düşük seviye diller ise hafızaya doğrudan müdahale ederler. (a=5 atadığımızda bilgisayar gider onu RAM a yazar a değerini çağırdığında gider onu çağırır ekrana getirir)

Backend ---java

Oyun , işletim sistemi, profosyonel uygulamalar istiyorsanız-----C++

Veri birimi yapmak ist, oyun, yapay zeka, backend --------- Python

Fronted yani ön yüz uyg yazmak istiyorsanız----javascript gibi programları kullanabilirsiniz.

Bunlar gibi onlarca dil var, ihtiyaca göre belirleyebilirsiniz.

Bunları kullanarak;

* Bir server uygulaması
* Bir web uyg
* Bir desktop, masaüstü uyg
* Bir mobil uyg yapabilirsiniz.

YSeviye---- performansları daha yavaştır. (Python işletim sistemine gidecek hafızaya gidecek

DS------İşlemci ram ini doğrudan kullandığı için performans fazladır.

YS------kodlar temizdir, basittir, her bakan anlar

DS----Diğerinde öyle değildir

YS -----Daha esnek, okunması kolay

DS--Üstün perf ancak yazması zor

YS-----Nesne tanımlıdır, fonksiyonel olarak çalışırlar (masa yı bir değişkende tutabilirsiniz)

DS-----Bu destek yoktur ve öğrenmesi zordur

YS-----Kominitisi çok fazladır (internette aradığınızı rahat bulabilirsiniz)

Makine Dili

1010101 lardan meydana geliyor, binlerce kod satırlardan meydana geliyor yazması burada çok zor. (bazı hacker ler burada binlerce 1 ler ve 0 larda arayıp bulup değişiklik yapabiliyorlar.)

Assembly dili

Burada 1 ler ve 0 lar yerine o kod guruplarına isimler takılmış (minemonik),

İki sayıyı toplamak için yaptığımız işlem bu mov, add, sub A,B

1. Bellek adresleri: Bellek adresleri, segment:offset şeklinde ifade edilir. Segment, bellek segmentini ve offset, bellek segmentindeki belirli bir konumu belirtir. Örneğin, [DS:0x1234] adresi, DS segmentinde 0x1234 konumundaki bellek adresine işaret eder.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Mov ½esp, ½ebp -------🡪 İşlemci içerisindeki saklama alanları yani register dediğimiz küçük bellekler, işte bu regester in ne olduğunu , komut yapısının ne olduğunu bilmeniz gerekir.

Bilgisayar register'ı, bilgisayarın işlem yapmak için geçici olarak verileri sakladığı bir tür yüksek hızlı veri deposudur. Register'lar, bilgisayarın ana bellekten daha hızlı erişebileceği küçük boyutlu bellek hücreleridir ve bilgisayarın işlemci yürütme hızını artırmak için kullanılır. Register'lar, işlemcilerin içinde bulunur ve birkaç bitlik boyutlara sahip olabilirler. Her register, belirli bir işlemci işlevini yerine getirmek için ayrılmıştır ve belirli bir amaca hizmet eder. Örneğin, bir kayıt belirli bir matematik işlemi gerçekleştirmek veya bir komut seti işleme verileri aktarmak için kullanılabilir. Bilgisayar register'ları, bilgisayarın işlem yapma yeteneğine ve performansına doğrudan etki eder.

Hala kullananlar var ama bu dil donanıma bağlı bir dil olduğu için çok zor oluyor, bir de bu her işletim sistemine göre değişiyor.

En hızlı kodu nasıl yazarım derseniz, bu dilde çok hızlı çalışır ama bu dilde yazmak çok çok zordur.

.asm uzantılı kodlar makine diline çevriliyor (.exe)

Ben kodu assebly dilinde de yazsam bilgisayarda yine makine diline çevrilir.

Yüksek seviye diller

YSDillere bakıldığında, günlük olarak kullandığımız terminolojileri kullanarak yazılmış dillerdir ama ingilizce diliyle yazılır. Amacımız komut ezberlemek değil, bazı özel kelimeleri bilmek ve bu kelimelerle (buna reserved---ayrılmış) komutları yazmak gerekiyor

İf, while, do --- tabi bunlar kullandığınız dile göre farklılık gösterebilir. Biraz konuşma dilindedir.

Yüksek seviye dilleri anlayabiliyorum işte bunlar kaynak kod oluyor neyin kaynak kodu assembly dilinin kaynak kodu.

Bu da insanın okuyabileceği programlama diline özgü metindir. İşte kaynak kod bu.

Siz source kodu yazıyorsunuz bunlardan birinden (compıler, interpreter) geçerek cpu ya gidiyor ve orada çalıştırılıyor.

Java-----Compiler( kodun tamaını alıp tek bir seferde makine diline çeviriyor) ----derleyici

Python------İnterpreter-----yorumlayıcı (tek tek alıp çeviriyor)-🡪

İkisinin de dezavantajları ve avantajları var.

Mesala indirip kurduğumuz programların çoğu compiler tabanlı çevriliyor, ticari olarak bunu yapıyorlar bize kodları vermemek için.

WEB tabanlı sistemlerde interpreter –yorumlayıcı kullanarak yapılıyor.

Fark şu java da çalışır komple edersiniz kod çalışır. Python da yaz kod yorumlanır, interpreter edilir Python kodu gönderir

Kaynak kod

Geliştiricilerin yazdığı, her hangi bir dilde yazılmış kodlara kaynak kod diyoruz.

Bizin anlayabileceğimiz yüksek seviyeli bir dille yazılmış koda kaynak kod diyoruz.

Bizim ilk kendi yazdığımız programımız kaynak kodumuzdur.

Microsof ne yapıyor programı yapıyor programı derliyor bize makine diline göre yazılmış halini gönderiyor.( .exe uzantılı olanı)

Bir çok uygulama bu şekilde yapılıyor, ticari amaçlı olarak bu yapılıyor.

OPEN SOURCE SOFTWARE

Kaynak kodu veren yazılımlara biz kaynak kodu diyoruz.

Bunun da öncülerinden biri LİNUX dür

Günümüzde de çok kullanılan bir işletim sistemidir.

Linux ü geliştirdiğinde bunun herkes tarafından geliştirilebilir bir kod olmasını istiyor, kaynak kodları internet aracılığıyla herkese sunuyor.

Linux işletim sistemi genellikle ücretsiz olarak sunulur. Linux, açık kaynak kodlu bir işletim sistemi olduğu için, kaynak kodu herkese açıktır ve kullanımı, dağıtımı ve değiştirilmesi ücretsizdir. Ancak, Linux işletim sistemiyle birlikte gelen bazı özellikler veya dağıtımlar için ücretli lisanslar mevcut olabilir. Örneğin, bazı Linux dağıtımları, özel destek veya ticari kullanım için ücretli lisanslar sunabilir. Ancak, çoğu durumda, Linux işletim sistemi tamamen ücretsizdir ve isteyen herkes tarafından indirilebilir ve kullanılabilir.---------------------------------------------------------------------------------

Ancak, bazı Linux dağıtımları veya paketleri için ticari destek veya ek özellikler sunan şirketler, ücretli lisanslar veya abonelikler sunabilirler. Örneğin, Red Hat Enterprise Linux (RHEL) ve SUSE Linux Enterprise Server (SLES), ticari kullanım için özel olarak tasarlanmış Linux dağıtımlarıdır ve bu dağıtımlar için ücretli lisanslar veya abonelikler satın almak gerekebilir. Bu lisanslar, ticari müşterilerin özel destek, güvenlik güncellemeleri ve ek özellikler gibi hizmetlere erişim sağlamasına olanak tanır.

Ayrıca, bazı özgür ve açık kaynaklı yazılımların kullanımı da ticari olarak kısıtlanmış olabilir. Örneğin, bazı özgür yazılımların kullanımı, ticari kullanım için özel bir lisans gerektirir. Bu lisanslar, yazılımın kaynak kodunu kullanma, dağıtma ve modifiye etme haklarını sınırlayabilir.

Sonuç olarak, Linux işletim sistemi genellikle ücretsizdir ve herhangi bir ücret talep edilmez. Ancak, ticari destek veya ek özellikler sunan bazı şirketler, özel lisanslar veya abonelikler sunabilirler. Ayrıca, bazı özgür yazılımların kullanımı da ticari olarak sınırlı olabilir.

Dolayısıyla, herkes onun geliştirmiş olduğu kodları alıp, kendi kodlarıyla güncelleyip yeni yeni versiyonlar, yeni yeni sürümler geliştirebiliyorlar.

İşte bu gibi sistemlere biz open source diyoruz.

Bunlar da diğer açık kaynak kodlu yazılımlardır

Apache

Mozilla Firefox

LibreOffice

PostgreSQL

WordPress

İnternetin gelişmesiyle bazı insanlar açık kod paylaşmaya başladılar, (önce moda oldu sonra standart oldu. )

Bu insanların önderliğinde programı yazıyorlar, kaynak kodunuda internete koyuyorlar buna bakın kimseninkini çalıyor muyum kontrol edin bakın hatta alın geliştirin kendiniz kini yazın diyorlar.

Linux ün kodu internette açık olarak var----- bunu alıp değişiklik yapıp, karşılaştırıp kendi Linux ümüzü yapabiliriz.

Free software----- yapıyorlar tamamen ücretsiz olarak paylaşıyorlar(etik nedenlerle- özgürlükçü bir yaklaşımla paylaşıyorlar.)

Burada kodu paylaşmıyorlar, programın uygulamasını paylaşıyorlar

Free---biz daha fazla özgürlük istiyoruz (etik değerler)

Open source----biz de daha yüksek kalite isteriz diyor. Biz aslında kullandığımız ürünün içini görmek isteriz, hatta duruma göre geliştirmek isteriz

Open source -----daha iyi bir kalite elde etmek için, herkes katkıda bulunsun….(teknik yaklaşım)

Bug duymayan var mı ?

Nedir

İngilizce anlamı böcek,

Bilgisayardaki hatalardır. Uygulamalardaki, yazılımlardaki hatalara biz bag diyoruz.

Bu hataları ayıklama işlemine de dibag deniyor.

* Programdaki kısır döngü bug dır
* Siz ekrana 15 yazdırmak istersiniz 16 yazdirirsınız
* Siz suyun sıcaklığını kontrol etmek yerine oda sıcaklığını kontrol ediyorsanız bu da bir bug dır arkadaşlar

Peki bu bug ların hayatımızdaki anlamları ne olacak

İki tip bug vardır

1. Yazım hatası (sintax)
2. Mantık hatası olabilir (suyun sıcaklığı, kısır döngü)

Buradaki bug ları bulup düzeltebilir misiniz

Kütüphane / Paket / Frimwork

Kütüphane ; kodların bir arada bulunduğu ortam. Daha önceden yazılmış yazılım kütüphaneleri.

Mesela (tamir kitabı , matamatik ödevi) (import ederiz----import pandas : içinde verileri hızlıca hesaplamak için herşey önceden yazılmıştır başkaları tarafından, matematik kütp için math yükler

dosyaya yazdırmak istersek operating sistem kütüphanesi yükler….)

Kütüphane ----- Daha önceden yazılmış kod parçacıkları, sınıflar, prosedürler, scripler, konfigürasyon verileri ve ayar dosyaları olabilir , bunların tamamı tek bir kılıf altında toplanmış haline layböry denir.

Bunlar bir develop tarafından yazılır bu kütüphanelere eklenir biz de bu sayede bunları da yazmak için ekstra uğraşmak yerine bunları hazır olarak kullanıyoruz. İşte bu kütüphanelerin böyle güzel bir işlevi var.

Herkes yapabilir, çokça yazılan işlemleri alt görev haline getiriyor. Mesela siz toplama işlemi yapacaksınız bunları alt program, alt görev haline getiriyor. Bunlar a fonksiyon haline getiriyorsunuz. Fonksiyonlar birleşip bir modülü meydana getiriyor.

Hesaplama modülü, yazı yazma modulu, gibi modüllerde birleşip pekiç haline geliyor.

Pekiçlar birleşiyor bunlarda kütüphaneleri oluşturuyor. Burada istadiğiniz fonk, modülü veya pekiç kullanabilirsiniz siz yeterki bu kütüphaneyi programınıza ekleyin.

(türev alma gibi

İmport math

Math.integral(x^2,0,1)

Bunların amacı neymiş, bir işlemi kod yazmadan otomatikleştirmek için bir programa manuel olarak yazılım kitaplığı ekleyebilirsiniz.

Her dilin kendine göre kütüphanesi var, ücretsizdir.

Paket-----🡪 kütüphanenin küçük parçalara ayrılmış hali

Fremawork --🡪 geliştiricilerin uygulama yapmak için kullandığı bir yazılımdır.

Kütüphaneler esnektir, istediğimiz programı, istediğimiz fonk içinde kullanabiliriz. Frimwörk öyle değildir bizi yönlendirdiği şekilde yapmamız gerekiyor yoksa çalışmıyor hata veriyor. O şekilde yazmaya bizi zorluyor. belirli bir kodlama standardına uyulmasını sağlar.

Dezavavntajı var evet ama avantajı sizin en iyi şekilde kod yazmanızı garanti ediyor.

Burada da PHP dili ile alakalı kütüphaneler ve frameworkler var, PHP bir backend dilidir, serverlarda uygulama yazmanın dilidir. Eski ama hala kullanılıyor.

Bunları kullanıyorsunuz, server dili uygulamış oluyorsunuz.

Sunucu tarafı programlama, bir web uygulamasının kullanıcı tarafından görünen arayüzünün ötesinde gerçekleşen işlemleri yönetir. Örneğin, bir kullanıcının bir web sitesindeki bir formu doldurup göndermesi durumunda, sunucu tarafı programlama, bu form verilerini işleyerek veritabanına kaydetme veya kullanıcının talep ettiği işlemleri gerçekleştirme gibi işlemleri gerçekleştirir.

Fremawork deki en önemli şey hayatı kolaylaştırmak, bug ları azaltmak ve kısa sürede çalışan uygulamaları üretmektir.

FRONTED / BACKEND

Fronted bir programın görünen kısmının olduğu yazılımdır.

Backend ise işin arka tarafında işlemlerin ve hesaplamaların yazıldığı yazılımdır.

Kullanıcının görmediği ; sizin verinin işlendiği, arka planda nasıl işlendiği size nasıl verinin iletildiği bunlar hep arka plandadır.

Bir Fronted geliştiricisi olacaksanız eğer bilmeniz gereken bazı temel diller vardır, nedir bunlar;

Html -----içerik oluşturmak için

Css -----görsellik oluşturmak için

Java script---- bu staylinge hareket vermek için yani siz bir uygulamada bir yere tıklayacaksınız sayfa şöyle açılsın stok durumunu güncellesin gibi işlemler için de bu dil kullanılır.

Stayling için CSS, veri girişi için JavaSC bilmek gerekiyor.

"Styling" veya "style", bir web sitesinin veya uygulamasının görünümünü ve hissini belirlemek için kullanılan tekniklerin tümüdür. Styling, bir web sayfasının tasarımını belirleyen renkler, yazı tipleri, arka planlar, çerçeveler, düzenler ve diğer öğeleri içerebilir.

Styling, HTML ve CSS kullanılarak gerçekleştirilir. HTML, bir web sayfasının yapısını tanımlayan kodlamadır, CSS ise bir web sayfasının görünümünü ve hissini belirleyen stil bilgilerini tanımlayan bir dil olarak kullanılır.

Javascript dilinin de kullandığı kütüphane ve frimwörkler vardır

React------libr

Vue-----FW

Angubr ---FW

Bu gibi kütüphane ve fremaworklerle çok daha modüler çok daha hızlı bir şekilde bir uygulama yazmak mümkündür.

Bacend e gittiğinizde burası için çok alternatif var; PHP, JAVA, Go, PYTHON

Bu gibi dillerde bekend uygulaması yapabilirsiniz. Bunlardan birini kullanmanız yeterlidir. Birini bilince diğerini öğrenmek çok zor değildir.

Bekend ve Frontad karşılaştırılması

Fronted ----görünen kısmı

Bekend ise görünmeyen kısmıdır

F---Kullanıcı tarafını işaret eder

B---Server tarafını işaret eder

F---web sitesinin kullanıcı tarafından görünen tarafıdır

B---Mutfakta olan bütün işlerdir

F-----Genelde görünür bölümleri içerir

B---Veri tabanı işleri, hesaplamaları ve diğer bölümleri içerir

F---- kullanıcının etkileşime girdiği alandır

B---- etkileşime arka tarafındaki beyindir, mantıktır

En temel uygulamalar

Software hakkında aklınızda kalan şeyi yazar mısınız