

Bir bilgisayar programı nasıl çalışır?

Bir program bilgisayarı kullanılabilir hale getirir. Program olmadan, bir bilgisayar, hatta en güçlü olanı bile, bir nesneden başka bir şey değildir. Benzer şekilde, çalanı olmadan bir piyano, ahşap bir kutudan başka bir şey değildir.

Bilgisayarlar çok karmaşık görevleri yerine getirebilir, ancak bu doğal olarak gelen bir yetenek değildir. Bilgisayarın doğası oldukça farklıdır.

Yalnızca son derece basit işlemleri gerçekleştirebilir; örneğin, bir bilgisayar karmaşık bir matematiksel işlevin değerini kendi başına değerlendiremez, ancak bu yakın gelecekte olasılık alanlarının ötesinde değildir.



Çağdaş bilgisayarlar yalnızca toplama veya bölme gibi çok temel işlemlerin sonuçlarını hesaplayabilirler, ancak bunu çok hızlı yapabilirler ve bu işlemleri her zaman ve bir çok kere tekrarlayabilir.

Uzun bir yolculukta ulaştığınız ortalama hızı bilmek istediğinizi düşünün. Mesafeyi biliyorsunuz, zamanı biliyorsunuz, hıza ihtiyacınız var.

Doğal olarak, bilgisayar bunu hesaplayabilir, ancak bilgisayar mesafe, hız veya zaman gibi şeylerin farkında değildir. Bu nedenle, bilgisayarı şu şekilde yönlendirmek gerekir:

- mesafeyi temsil eden bir sayıyı al;
- seyahat süresini temsil eden bir sayıyı al;
- ilk değeri ikincisine böl ve sonucu bellekte sakla;
- sonucu (ortalama hızı temsil eden) okunabilir bir biçimde görüntüle.

Bu 4 basit aksiyon **programı** oluşturur. Tabii ki, bu örnekler bilgisayarın anlayabileceği bir forma dönüştürülmemiştir ve bilgisayarın anlayabileceğinden çok uzaktır, ancak bilgisayarın kabul edebileceği bir dile çevrilecek kadar iyi durumdadır.

Dil anahtar kelimedir.

Doğal diller ve programlama dilleri

Dil, düşünceleri ifade etmek ve kaydetmek için kullanılır (ve bir araçtır). Etrafımızda birçok dil var. Bazıları beden dili gibi ne konuşma ne de yazma gerektirmez; en derin duygularınızı tek kelime etmeden çok hassas bir şekilde ifade etmek mümkündür.

Her gün kullandığınız başka bir dil, iradenizi göstermek ve gerçekliği düşünmek için kullandığınız anadilinizdir. Bilgisayarlar da kendi dillerine sahiptir, bu dil **makine** dili olarak adlandırılır ve çok ilkelidir.

Teknik olarak çok gelişmiş olsa bile, bir bilgisayarda herhangi bir zeka izi yoktur. İyi eğitilmiş bir köpek gibi olduğunu söyleyebilirsiniz - sadece önceden belirlenmiş bir dizi bilinen komuta yanıt verir.

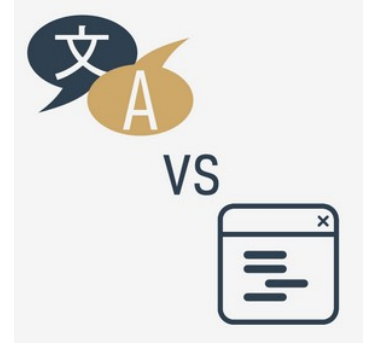
Tanıdığı komutlar çok basittir. Bilgisayarın “bu sayıyı al, diğerine böl ve sonucu kaydet” gibi emirlere yanıt verdiğini düşünebiliriz.

Bilinen komutların tam setine, bazen **IL** olarak da kısaltılabilir, bir **talimat listesi** denir. Farklı bilgisayar türleri, IL 'lerin boyutlarına bağlı olarak değişebilir ve talimatlar farklı modellerde tamamen farklı olabilir.

Not: Makine dilleri insanlar tarafından geliştirilmiştir.

Hiçbir bilgisayar şu anda yeni bir dil oluşturamıyor. Ancak bu yakında değişebilir. Öte yandan, insanlar da çok farklı diller kullanır, ancak bu diller doğal olarak gelişmiştir. Dahası, bu diller hala gelişmeye devam ediyor.

Her gün yeni kelimeler ortaya çıkıyor ve eski kelimeler kayboluyor. Bu dillere **doğal diller (natural languages)** denir.



Bir dili, dil yapan şey nedir?

Her dilin (makine veya doğal dil , farketmez) aşağıdaki unsurlardan oluştuğunu söyleyebiliriz:

- bir **alfabe**: belirli bir dildeki kelimeleri oluşturmak için kullanılan bir dizi sembol (örneğin, İngilizce için Latin alfabesi, Rusça için Kiril alfabesi, Japonca için Kanji vb.)
- bir **lexis**: (sözlük olarak da bilinir) adilin kullanıcılarına sunduğu bir dizi kelime (örneğin, "computer" kelimesi İngilizce sözlükten gelirken "cmoptrue" değildir; "chat" kelimesi hem İngilizce de hem de Fransızca sözlük de mevcuttur, ancak anlamları farklıdır)
- bir **syntax (sözdizimi)**: belirli bir kelime dizisinin geçerli bir cümle oluşturup oluşturmadığını belirlemek için kullanılan bir dizi kural (resmi veya gayri resmi, yazılı veya sezgisel olarak hissedilir) (örneğin, "I am a python" sözdizimsel olarak doğru iken, "I a python am" değildir.)
- **semantics (anlambilim)**: belirli bir cümlenin anlamlı olup olmadığını belirleyen bir dizi kural (ör. "Bir çörek yedim" anlamlıdır, ancak "Bir çörek beni yedi" anlamsızdır)

IL , aslında **makine dilinin alfabesidir**. Bu, bir bilgisayara komutlar vermek için kullanabileceğimiz en basit ve en temel sembol kümeleridir. Bilgisayarın ana dilidir.

Ne yazık ki, bu dil insan anadilinden uzak bir dildir. Hem bilgisayarların hem de insanların başka bir şeye, bilgisayarlar ve insanlar için ortak bir dile veya iki farklı dünya arasında bir köprüye ihtiyacı vardır.

İnsanların programlarını yazabilecekleri bir dile; bilgisayarların da programları yürütmek için kullanabileceği bir dile ihtiyaçları vardır. Makine dilinden daha karmaşık ancak doğal dilden daha basit bir dile...

Bu tür dillere genellikle üst düzey programlama dilleri denir. İnsanlar tarafından okunabilen semboller, kelimeler ve sözleşmeler kullandıkları için en azından doğal dillere benzerler. Bu diller, insanların IL'ler tarafından sunulanlardan çok daha karmaşık olan bilgisayarlara komutları ifade etmelerini sağlar.

Üst düzey bir programlama dilinde yazılmış bir programa **kaynak kodu** denir (bilgisayarlar tarafından yürütülen makine kodunun aksine). Benzer şekilde, kaynak kodunu içeren dosyaya **kaynak dosya (source file)** adı verilir.

Derleme ve Yorumlama

Bilgisayar programlama, seçilen programlama dilinin öğelerini istenen etkiyi yaratacak sırayla oluşturma eylemidir. Etki, her özel durumda farklı olabilir - programcının hayal gücüne, bilgisine ve deneyimine bağlıdır.

Tabii ki, böyle bir kompozisyon birçok anlamda doğru olmalıdır:

- **alfabetik olarak** - bir programın, Roma, Kiril, vb. gibi tanınabilir bir alfabeye yazılması gerekir.
- **sözlük olarak**- Her programlama dilinin sözlüğü vardır ve onda ustalaşmanız gerekir; Neyse ki, herhangi bir doğal dilin sözlüğünden çok daha basit ve daha küçüktür;
- **söz dizimsel olarak** - Her dilin kuralları vardır ve bu kurallara uyulmalıdır;
- **semantik (anlamsal) olarak** - programın anlamlı, mantıklı olması gerekir.

Ne yazık ki, bir programcı yukarıdaki dört unsurun her birinde hata yapabilir. Bu unsurların herhengi birinde hata programın tamamen işe yaramaz hale gelmesine neden olabilir.

Başarıyla bir program yazdığınızı varsayalım. Bilgisayarı bu programı çalıştırmaya nasıl ikna edeceğiz? Programınızı makine diline dönüştürmeniz gerekiyor. Neyse ki, bilgisayarın kendisi tarafından bu çeviri yapılabilir, bu da tüm süreci hızlı ve verimli hale getirir.

Bir programı **üst düzey bir programlama dilinden makine diline dönüştürmenin iki farklı yolu vardır**:

DERLEME (COMPILATION)- kaynak program bir kereliğine makine kodunu içeren bir dosyaya çevrilir (örneğin, kod MS Windows altında çalıştırılacaksa bir .exe dosyası) (ancak, kaynak kodunu her değiştirdiğinizde bu işlem tekrarlanmalıdır); bu dosya dünya çapında dağıtılabilir durumdadır. Bu dönüşümü gerçekleştiren programa derleyici veya çevirmen denir;

YORUMLAMA (INTERPRETATION) - kaynak program her çalıştırılması gerektiğinde, programın kullanıcısı tarafından makine diline çevirebilir; bu tür bir dönüşümü gerçekleştiren programa yorumlayıcı denir. Çünkü kod, her çalıştırılmak istendiğinde yorumlanmaktadır. Bu aynı zamanda kaynak kodunu olduğu gibi dağıtamayacağınız anlamına gelir, çünkü son kullanıcının bunu yürütmek için yorumlayıcıya da ihtiyacı vardır.

Bazı temel nedenlerden dolayı, belirli bir üst düzey programlama dili bu iki kategoriden birine girmek üzere tasarlanmıştır.

Hem derlenebilen hem de yorumlanabilen çok az dil vardır. Genellikle, bir programlama dili kurucularının zihninde bu faktörle yansıtılır - derlenecek mi yoksa yorumlanacak mı?