

Bilgisayar Mühendisliği Eğitiminde Özgürlük

Dr. Onur Tolga Şehitoğlu

ODTÜ Bilgisayar Mühendisliği Bölümü Öğretim Üyesi/LKD Yönetim Kurulu Üyesi
onur@ceng.metu.edu.tr

Kullanım, paylaşma ve geliştirme özgürlüğünü kullanıcıya veren ve dünya çapındaki birçok gönüllü tarafından geliştirilen özgür yazılımların bilgisayar dünyasında oynadığı önemli rol, bilgi teknolojileri konusunda profesyonel olmak üzere eğitim gören birçok branşta ama en çok Bilgisayar Mühendisliği eğitiminde çok daha kritik ve vazgeçilmezdir. Bilgisayar bilimleri ve mühendislik eğitimi veren kurumlar ve özgür yazılımlar, özgür yazılımların ve hatta yazılımların ortaya çıktığı ilk dönemden beri birbirleriyle çok yakın temasta olmuşlardır. Özgür yazılımların tarihine baktığımızda akademinin özgür yazılıma, özgür yazılımların eğitime verdiği katkıların çok büyük olduğunu görüyoruz.

BAŞTA BÜTÜN YAZILIMLAR Özgürdü

Bilgisayarların ilk dönemlerinde yazılımın bir mal olarak düşünülmediği, bilgisayarların daha iyi çalışması için bilgisayarla beraber verilen aksesuar olarak görüldüğü bir dönem var. Bu dönemde yazılım bugünkü özgür yazılım ekipleri gibi kullanıcı grupları tarafından geliştiriliyor. Bilgisayar bilimlerinin en bakir araştırma alanlarından olduğu dönemde yazılım dünyasındaki birçok yeniliğin üniversite ile yana geliştiğini görebiliriz. Var olan yazılımların öğrenciler için bir deney ortamı olduğu, öğrencinin görerek öğrendiği, denediği, ürettiği ve endüstrinin de akademik katkılarla yazılımı geliştirdiği bu dönem 70'lerin sonuna kadar devam etti. Çağdaş işletim sistemlerinin çoğunun

örnek aldığı Unix işletim sisteminin de ilk 7 sürümü bu şekilde kaynak kodu açık olarak, üniversite ve diğer kurumlarla paylaşılarak gelişti. Daha sonraki sürümleri kapatılan Unix'in 7. sürümü daha sonra BSD gibi üniversite ortamında gelişen işletim sistemlerinin temelini oluşturdu.

Unix'in 7. sürümünden sonra yazılımın yavaş yavaş bilgisayardan ayrı olarak satılabilir mal haline geldiği dönemde şirketler programları kaynak kodları olmadan dağıtmaya, kopyalama ve kullanım haklarını kısıtlamaya başladılar. Kuşkusuz bu gelişmelerden en kötü etkilenebilecek kurumlar Bilgisayar Bilimleri ve Bilgisayar Mühendisliği eğitimi veren okullardı. Çünkü çalışan bir yazılımın içine bakma şansı yitirilmekte, deney ortamları daralmakta ve kaybolmaktaydı. 80'li yılların sonunda yazılım firmaları bunla yetinmeyip derleyicileri de işletim sisteminden ayrı satmaya başladılar. Derleyici olmadan eğitimlerini sürdüremeyecek bu kurumların donanım ve işletim sistemine ek olarak geliştirme ortamına da para vermek zorunda kalması başka bir talihsizlik olacaktı.

Fakat bu korkunç tablo hiçbir zaman bu şiddette gerçekleşmedi. Çünkü üniversiteler yazılım üretme ve ürettiğini paylaşma geleneğini sürdürdü. Unix 7. sürüm üzerine kurulu işletim sistemleri, programlama dili derleyici ve yorumlayıcıları ve kütüphaneler üniversiteler tarafından hem eğitimin parçası olarak, hem de gereksinimden

dolayı geliştirildi ve paylaşıldı. Üniversiteler doğal olarak 80'li yıllarda başlayan özgür yazılım hareketinin vazgeçilmez destekçisi ve kullanıcısı oldular.

Eğitim ve Kaynak Kod

Yıllar önce bir gazetenin köşesinde bana biraz komik de gelen bir haber görmüştüm. Bir veterinerlik fakültesinde yaralanan bir ineğin ameliyat yerini kapatmak yerine bölgeye saydam bir pencere dikmişlerdi. Yaşamakta olan ineğin iç organları küçük bir pencereden gözlenebiliyordu. Bize belki itici gelebilecek bu manzara, aslında veterinerlik öğrencileri için yaşayan organları gözlemlemek açısından belki de kaçınılmaz bir fırsat olabilir.

İşte ben var olan profesyonel nitelikte yazılımların açık kaynak kodunu Bilgisayar Mühendisliği öğrencileri için şeffaf bir ineğe benzetiyorum. Bir sistemin, bir sunucunun, bir bileşenin, bir algoritmanın nasıl çalıştığı, ne kadar karmaşık olduğu, sorunlara yaklaşım, çözüm yollarını görmek, belki de organ tasarlamayan veterinerlerin organı görmelerinden çok daha önemli. Örneklerden öğrenme belki en hızlı öğrenim yöntemlerinden birisi. Yazılımların kaynak kodu öğrenciye karmaşık yazılım sistemlerinin ders kapsamındaki küçük örneklerle gösteremeyeceğiniz gerçek yaşam sorunlarının çözümlerini gösteriyor.

Örneğin bir işletim sisteminin kaynak kodu olmadan bir işletim sistemi dersini vermek, örnek ağ

uygulama kaynak kodları olmadan bilgisayar ağıları dersi vermek, birçok alanda örnek çözümler olmadan kavramların nasıl çözüleceğini gösterebilmek neredeyse olanaksız derecede yavaş olacaktır.

Kaynak kodun eğitimdeki diğer bir önemli faydası değiştirilerek gözlemlenebilmesi. Yani yazılım üzerinde deneyler yapma şansı. Bir kavramı sistemin davranışını sadece gözlemleyerek değil, sistemle etkileşerek kavrayabilirsiniz. Bir yazılım sisteminde etkileşimin gidebileceği en son nokta en düşük düzeyde, kod düzeyinde etkileşebilmektir. Dolayısıyla biz Bilgisayar Mühendisleri kaynak kodu görmenin ötesinde onu değiştirebilme özgürlüğüne gerek duyuyoruz. Bu özgürlüğü ticari yazılımlarda parasıyla dahi almanız hiç kolay olmayacaktır.

EĞİTİMDE YAZILIM GEREKSİNİMİ VE MALİYET

Yazılımların doğasına baktığımızda Bilgisayar Mühendisliği eğitiminde gereksinim duyacağımız yazılımların özelleşmiş, son kullanıcıya değil geliştiricileri hedefleyen derleyici, geliştirme ortamı, sonucu ve ileri düzey kütüphane gibi yazılımlar olduğunu söyleyebiliriz. Bu gruptaki ticari yazılımların alıcısı bilgi teknolojileriyle profesyonel olarak ilgilenen, bu yazılımlar aracılığıyla kâr elde etmeyi planlayan kişi ve kurumlardır. Dolayısıyla bu yazılımların hele eğitimde kullanılacak çeşitlilikte, birçok firmanın birçok ürün çeşidiyle elde edilmesi çok pahalıdır. Bunu sadece bilgisayar konusunda değil mimarlık ve mühendislik eğitiminde birçok konuya genelleştirebiliriz.

Diğer bir açıdan belli bir ticari ürünün üniversite tarafından erişilebilir olmaması, öğrencilerin bu ürünü tanımaması, mezun olunca da kullanmaması, uzun vadede kolay elde edilebilir bir ürüne göre ciddi bir dezavantaj olarak firmaya dönecektir. Büyük firmalar bu anlayışla genel olarak özellikle Bilgisayar Mühendisliği bölümlerine çok ciddi indirimler vermekte hatta bütün ürünleri ücretsiz olarak sunmaktadırlar. Hâlâ eğitim kurumlarından

yazılımlar için çok ciddi ücretler talep eden firmalar olsa da satın alma maliyetinin eğitimde kullanılacak yazılımlar konusunda (firmanın pazarlama anlayışının ne kadar esnek ve akılcı oluşuyla bağlantılı olarak) kısa vadede çok önemli olmadığını söyleyebiliriz.

Fakat daha stratejik düşündüğümüzde eğitimde ticari yazılım kullanmakla ilgili diğer maliyetleri görürüz. Öncelikle görebildiğimiz, değiştirebildiğimiz, üzerinde deney yapabildiğimiz kaynak kodundan vazgeçmiş oluruz. Bazı şirketlerin çıkarttığı "kaynak paylaşımı" türü lisanslar bize bu özgürlükleri vermemektedir. Kaynak koda rica minnet ulaşmış olsak bile değiştirme ve tekrar derleme şansımız yoktur.

Diğer bir konu ticari bağımlılıktır. Çoğu hibe ve benzeri anlaşma yanında müfredat değişikliği, müfredatın özelleşmesi, diğer yazılımları hariç tutması gibi beklentilerle gelmektedir. Bu beklentiler açıkça telaffuz edilmese de uzun vadede hibenin sona ermesi, yeni sürümler için yinelenmemesi gibi sonuçlar doğurmaktadır. Bağımlılık bir süre sonra, aksi durumda eğitimi aksatacak düzeye eriştiğinde ise artık şirketin sizden ticari beklentiye girmesi beklenen bir sonuç olur. Bugün birçok üniversite işletim sistemi ve kelime işlemci gibi özgür alternatifleri olan çok standart yazılımlara bağımlılıkları yüzünden ciddi kampüs lisans ücretleri ödemektedir. Zira birçok kurum gibi üniversiteler de "benim bu yazılımlara gereksinimim yok" diyemeyecek şekilde bir bağımlılık içine girmiştir. Yine ticari kaygılarla müfredatınızda kritik bir noktaya koyduğunuz bir ürünün desteği şirket tarafından sona erdirilebilir. Bu açıdan özellikle müfredatları ürünler üzerine değil kavramlar üzerine kurgulamak, uygulamalarda çeşitliliği ve alternatifleri olan konuları seçmekte fayda vardır. Özgür yazılımlarda bir proje sona erse bile kaynak kodu üzerinden projeyi devam ettirmek ve eğitimi aksatmamak mümkündür.

Ticari yazılımlar eğitimi daraltabilir. Şirketler yazılım geliştirme stratejilerini ticari kaygılarla yönlendirirler. Dolayısıyla eğitim

için çok kritik, bilimsel altyapıya yönelik bir konuda ticari yazılım bulmak olanaksız olabilir. Bulunan yazılımlarda da üniversiteler tek alıcı aldığı için tekelleşme ve yüksek maliyet gibi sorunlarla karşılaşılabilir. Örneğin yapay zeka gibi konularda vazgeçilmez olan fonksiyonel dillere ait geliştirme araçları özgür alternatifler olmasaydı bunları sadece yurt dışından yüksek maliyetle temin edebilecektik. Bu gibi konularda özgür yazılımların varlığı eğitimimizi bir anlamda kurtarmaktadır.

Bilgiyi Paylaşma Kültürü

Üniversiteler ve araştırma kurumları bilgiyi paylaşma kültürü açısından da özgür yazılımlarla doğal olarak birlikte olmak zorundadır. Bilimsel ürünler, araştırmaların sonucu elde edilen sonuçlar, yayınlanarak paylaşılmazsa bilimin büyümeyeceği, yozlaşacağı aşikârdır. Bu nedenle binlerce yıllık akademik öğretisi bilimsel araştırma sonuçlarının yayınlanması, kamuya açılması ilkesine dayanır. Özgür yazılım felsefesi de yazılımı bilgi olarak ele alır ve yazılım dünyasında gelişmenin daha hızlı olması amacıyla yazılım değiştirilebilir ve tekrar dağıtılabilir özgürlüğünü savunur.

Özgür Yazılımlar Vazgeçilmez

Bugün fiili durumda özgür yazılımlar bir eğitim kurumunun hemen elde edebileceği, kısa sürede derleyicisinden uygulama sunucusuna birçok karışık yazılım altyapısının ücretsiz olarak çalışır hale getirebileceği, geniş bir camiası olan, belgeleri erişilebilir (belgelendirmede eksiklik olduğunda kodun içine girilerek anlaşılabilir) eşsiz bir altyapı sunmaktadır. Uygulamalarda, kütüphanelerde, projelerdeki çeşitlilik eğitimin farklı yüzlerde, farklı yaklaşımlarla, farklı araçlarla yürüyebilmesini ve yazılımın özünden kavranabilmesini sağlar.

Bütün bunlar göz önüne alındığında Bilgisayar Mühendisliği eğitiminin özgür yazılımsız gerçekleşmesinin neredeyse olanaksız olduğunu, özgür yazılımın eğitime derinlik, açıklık, genişlik getirdiğini söyleyebiliriz.