

Bilgisayar mühendisliği eğitimi ve problemleri

Ahmet Babanlı

Bilgisayar Müh. Bölümü, Anadolu Üniversitesi, 26470 Eskişehir.

e-mail: ababanli@anadolu.edu.tr

tel.: 0222 321 3550 / 6555

Özet Mühendislik eğitiminin kalitesinin yükseltilmesi her zaman gündemde olan problemlerdendir. Kalitenin yükseltilmesi belli bir kriterler setinin oluşturulması ve uygulanması ile ilgilidir. Bu yazıda bilgisayar mühendisliğinin, özellikle onun yazılım dalının genel mühendislikte yeri incelenmiş, Türkiye üniversitelerinde bilgisayar eğitimi problemleri ele alınmış, eğitimin iyileştirilmesi kriterlerini oluşturmaya katkı sağlayabilecek konular tartışılmıştır.

Giriş

Son yıllarda Avrupa'da üniversitelerin toplumun taleplerini karşılamakta yetersiz kaldığı söylenmekte, üniversitelerin kalitesinin yükseltilmesine büyük ihtiyaç olduğu tartışılmaktadır. Üniversitenin kalitesi derken iki temel gösterge, eğitim ve araştırma göz önüne alınmakta, onların doğru orantılı olması düşünülmektedir. Bu göstergelerin birinin diğerine üstünlük sağlaması genel kalitenin düşmesine sebebiyyet verir fikiri yaygındır. Ama bazı araştırmalar onların birbiriyle direkt ilişkide olmadığını kanıtlamaktadır [1]. Mühendislik eğitiminin üniversite eğitiminde özel yeri vardır. Toplumun sürekli artan ihtiyaçlarının karşılanması için teknolojinin hızlı geliştirilmesi gerekmektedir ve burada mühendislik bilimleri baş rolü üstlenmektedir. Mühendisliğin lokomotifini ise bilgisayar mühendisliğidir. İzleyen bölümlerde mühendislik eğitimi ile ilgili problemler ele alınacak, bilgisayar mühendisliği eğitiminin genel durumu değerlendirilecek, ilgili konulara üniversitelerimizdeki bakış açıları incelenecek, bilgisayar eğitiminin iyileştirilmesine yönelik bazı öneriler gündeme getirilecek.

Mühendislik eğitimi

Mühendislik eğitiminin oluşturulması ve hızla gelişmesi XX yüzyıla denk gelmektedir. Gelişme sürecinde dört temel mühendislik dalı belirlenmiş ve okutulmaya başlanılmıştır: Elektrik, Mekanik, Kimya ve Sivil mühendisliği. Son 20-25 yılda endüstriyel gelişmenin ürünü olarak iki yeni mühendislik alanı onlardan ayrılmış ve bağımsız dal statüsü kazanmıştır. Bunlar Bilgisayar mühendisliği ve Bio-mühendisliktir [2]. Tarihsel olarak mühendislik eğitiminin temelini üç fen disiplini oluşturmaktadır: matematik, fizik ve kimya. Ama mühendisliği bu disiplinlerin sadece gelişmiş formu gibi algılamak yanlış olur.

Mühendislikle klasik bilimin arasındaki temel fark mühendisliğin toplumun bugün ve yarınki ihtiyaçlarını karşılamaya odaklanmış olmasıdır. Bir mühendis temel bilimlerin verdiği imkanlara dayanarak pratik problemleri ele alır, onun çözüm yollarını arar, uygulamalı metotlar ve stratejiler geliştirerek zaman, mekan ve ekonomi sınırlamaları ortamında problemin tam veya kısmi çözümüne ulaşmaya çalışır. Genel bilimsel araştırmacı ise genel çözüm (çok zaman teorik) peşindedir, onu şu anda pratik uygulama ilgilendirmez. Amaçların farklılığı eğitim programlarının da farklı olmasına getirmektedir. Öğretimin metodolojisi, disiplinlerin öğrenilme derinlikleri ve detayları ve genel pedagoji ders programlarının temellerini oluşturacaktır. Bu temelleri inceleyelim. Mühendisliğin metodolojisi derken biz genelde bir problemin (sistemin, cihazın vs) öyle modelinin hazırlanma prosesini göz önüne alırsak ki onun dolayısıyla tüm bileşenlerin davranışını değerlendire bilelim, tam olmayan, yetersiz, hatalı veriler ortamında tatmin edici çözüme ulaşabilelim, problemin yönetilebilirliği ile ilgili fikir sahibi olalım. Metodolojinin uygulanması ders programları dolayısı ile yerine getirilmektedir. Ders programlarının hazırlanmasında disiplinlerin bilimsel içeriklerinin yanısıra uygulanacak mekânın yerel özellikleri, programı hazırlayanların eğitime bakış açısı, ekonomik ve etik problemler gibi faktörler de göz önüne alınmalıdır. Bu faktörlerin doğru orantılanması mühendislik eğitimi metodolojisinin başarısını belirler. Uzmanlık çağdaş mühendislik eğitiminin temel stratejisidir. Farklı mühendislik programları mevcuttur. Aynı veya benzer disiplinlerin öğrenilmesi derinlik ve detaylılık açısından bu programlarda aynı değildir ve olmalıdır. Örneğin, "Diferensiyel denklemler" dersi mühendislik dallarında çoğunlukta derin öğrenilmesi gereken bir disiplindir. Ama hepsinde değil. Yeni mühendislik dallarının (bio-

mühendislik, nano-teknoloji gibi) ders programlarında genel mühendisliğin altyapısını oluşturan matematik-fizik-kimya listesi değiştirilmeli, listeye yeni dersler (biyoloji, bilgisayar bilimleri, tıp bilimleri) eklenmeli, derslere in öğrenilme derinlikleri güncelleştirilmelidir.

Ders programlarının uygulanmasında beklenen başarıya ulaşılması genelde kullanılan pedagoji stratejilerine bağlıdır. Geleneksel pedagoji yöntemi yüze-yüz metotudur: öğretim üyesi konuyu sınıfta bir grup öğrenciye belli zaman süresinde anlatır, yardımcı araç olarak yazı tahtasını ve/veya diğer çağdaş teknik cihazları kullanır. Çoğunlukta dersin temelini teorik kavramlar oluşturur. Teorik kavramların benimsenmesi ve pratik alışkanlığın elde edilmesi için ise seminerler ve laboratuvar çalışmaları yapılır. Çağdaş teknolojinin sunduğu imkanların kullanılması (Matlab tipi yazılım sistemleri, yerel bilgisayar ağları, İnternet) öğretim-öğrenim prosesini hızlandırır, kalitesini yükseltir. Bazı konuların öğrenilmesi öğrencinin serbest çalışmasını ister, diğerleri ekip çalışmasını faydalı bulur. Pedagojinin görevi bu süreci organize etmek, uygun eğitim metotlarını kullanarak onun verimli şekilde yürütmesini sağlamaktır.

Bilgisayar mühendisliği

Bilgisayar mühendisliği diğer mühendislik dallarına nazaran daha gençtir, temelleri tam olarak oturmuş değildir ve gelişme sürecindedir. Üç temel ana bilim dalı vardır: bilgisayar bilimleri, bilgisayar donanımı ve bilgisayar yazılımı. Örnek olarak bilgisayar yazılımını bir az detaylı inceleyelim. Yazılım mühendisliği daha hızlı gelişen, yaşamımızın demek olur ki tüm alanlarında kullanılan ve kullanılmasına ihtiyaç duyulan ve böylece piyasada sürekli istenilen bir uzmanlık dalıdır. Bu daha fazla ve kaliteli yazılım mühendislerinin hazırlanması gereksinimini ortaya koymaktadır. Ve bu gereksiniminin yakın gelecekte azalacağı veya sabitleşeceği gözlenmemektedir.

Bazı uzmanlar yazılımın geleneksel mühendislikle bir ilgisinin olmadığı fikrindedir. Ama yazılımın mühendisliğin özel bir dalı olduğu düşüncesi daha yaygındır. ACM/IEEE'nin hazırladığı öğretim projesinde [3] tüm mühendislik disiplinleri için (yazılım mühendisliği dahil) yedi temel özellik belirlenmiştir. Bunun yanı sıra yazılım mühendisliğini diğer mühendisliklerden farklı kılan iki önemli özelliğin olduğu da vurgulanmaktadır [4]. Birincisi şudur: yazılım ürünü soyuttur, diğer mühendislik ürünleri ise somuttur, fizikseldir. İkincisi, yazılım ürünü kesikli (discrete) temele dayalıdır, diğerleri genellikle sürekli (continuous) temeldir. Duyumsanabilirlik doğası olmayan yazılım ürünü insanın alışık olduğu görünebilirlik, değişebilirlik, doğa yasalarına

uyumluluk gibi özellikleri negatif yönde etkilemektedir. Biz ürünü görmek, ona dokunmak, iç yapısını anlamak istiyoruz. Yazılım ürünü genelde bunlara imkan tanımamaktadır, onun fiziksel dünya ile direkt ilişkisi yoktur.

Ama yazılım ürününün bazı avantajları da vardır. Örneğin, yazılım ürünü kolayca değiştirilebilir, kaynak kodu değiştirmeden ona yeni bileşenler eklenebilir, gereksiz parçalar çıkartılabilir, bir uygulama ortamından diğerine kolayca taşınabilir vs. Fiziksel ürünlerde bu tür değişiklikler zordur veya mümkün değildir: bir köprünün taşıma özelliğini yükseltmek istersek o köprü yıkılır ve yenisi inşa edilir.

Geleneksel mühendisliğin altyapısını sürekli matematik ve temel fen bilimleri (fizik, kimya) oluşturmaktadır, bilgisayar mühendisliği ise kesikli matematik ve bilgisayar bilimlerinden (programlama dilleri, algoritma teorisi, yapay zeka vs) yükümlü bilgilere dayanmaktadır. Bu önemli farklılıklar bilgisayar mühendislerinin hazırlanması prosesinin farklı incelenmesi ve organize edilmesi gereksinimini ortaya koymaktadır.

Üniversitelerimizde bilgisayar mühendisliği eğitimi

Bilgisayar mühendisliğiyle ilgili problemleri tartışmak amacıyla 90. yıllardan başlayarak Türkiye Bilişim Vakfı tarafından yılda iki defa olmak üzere Bilgisayar mühendisliği bölüm başkanları toplantıları organize edilmiş ve bugüne kadar kesintisiz yapılmıştır. Toplantılarda önemli problemler ele alınmış, bilgisayar eğitiminde olan eksikliklerin giderilmesi için tavsiyeler hazırlanmıştır. Bu toplantıların en önemli bir tane ürünü türkçe "Bilgisayar bilimleri ve mühendisliği dergisi"nin yayınlanmasıdır. Ama diğer önemli problemlerin çözümü bu kadar şanslı olmamıştır. Sebep, gereken uygulama mekanizmalarının oluşturulamaması veya uygulanamamasıdır. Biz burada bu toplantıların pratik sonuçlarının neden bu kadar az olduğunu tartışmayacağız, ama üniversitelerimizde bilgisayar mühendisliği eğitiminde olan bazı problemleri ele alacağız.

Ders programları

Ders programlarının hazırlanmasında göz önüne alınması gereken iki temel husus vardır: uluslararası standartlar (resmi veya resmi olmayan) ve yerel koşullar. Bunlardan birincisi eğitimin çağdaş seviyesinin sağlanmasına hizmet eder, ikincisi bu seviyeye ulaşmak için gereken stratejilerin oluşturulmasına yardımcı olur. Üniversiteyi kazanmış türk öğrencilerinin fen disiplinleri altyapısının (özellikle genel matematiğin) zayıf olduğu malum. Ama bilgisayar mühendisi için bu altyapının güçlü olması şart. Bilgisayar mühendisinin eğitimde elde etmesi gereken en önemli araç algoritmik düşünme tarzıdır. Bunun

için ise yeterli matematik altyapısının (özellikle teorem isbatlama başarısının) ve genel felsefe bilgisinin (düşünme, mantıksal çıkarım) olması gerekmektedir. Bunların üniversite önceki okullarda öğretilmesi yeterli değildir. Doğrudur, üniversite ders programlarında genel matematik geniş temsil edilmiştir. Ama onun öğretiminin metodolojisi standarttır ve bilgisayar mühendisinin istediği biçimde değildir: sürekli matematik anlatılıyor, uygulama basit matematik problemlerin çözümü ile sınırlıdır, teorem isbatlama azdır. Diğer önemli faktör öğretim üyesinin çoğunlukta matematik bölümünden olmasıdır. O, matematiği tüm öğrencileri aynı formda anlatıyor, onun için öğrencinin özelliği önemli değildir. Burada çözüm her bir bilgisayar bölümü kadrosuna bir matematik hocasının alınmasında olabilir (bazı bölümlerde bu pratik uygulanmaktadır). Bilgisayar mühendisliği ders programlarının içeriği ile ilgili şunların tartışılmasında fayda vardır.

1. “Diferensiyel denklemler” dersi diğer mühendislik dalları için çok önemlidir, ama bilgisayar mühendisliği için aynısını söylemek doğru olmaz. Onun ders programlarından çıkartılması ve konu ile ilgili kısa bilginin genel matematik derslerinde verilmesi daha uygun olur. Yerine ise bilgisayar mühendisliği için çok önemli olan “Graf teorisi “ dersi konula bilir. Graflarla ilgili kısa bilgiler birkaç farklı derslerde sistemsiz olarak verilmekte, aynı konular çok vakit tekrarlanmaktadır.
2. “Veri yapıları ve algoritmalar” dersi çoğunlukta tek dönemlik ders olarak geçer. Öğrencinin program geliştirme alışkanlığının biçimlendirilmesinde bu dersin öneminin büyük olduğu gerçeğini göz önüne alırsak bir dönemin yetersiz olduğunu görürüz. Bu dersin en az iki dönem zorunlu olarak verilmesi gerekmektedir (bazı üniversitelerde zaten böyledir). Yazılım mühendisliği öğrencileri için ise üçüncü dönem de öngörülebilir (zorunlu ve örneğin, “Karmaşık algoritma analizi” ismiyle).
3. Son yıllarda bazı bilgisayar mühendisliği bölümlerinde “Yapay zeka” dersi zorunlu dersler listesinden çıkartılmıştır (tam olarak çıkartılmış veya seçmeli ders olarak belirlenmiştir). Yapay zeka problemleri bilgisayarların meydana çıkması zamanından bugüne kadar bilim ve teknolojinin çözmeğe çalıştığı öncelikli problemler setinin zirvesinde yer almaktadır ve yakın gelecekte de bu durumun değişmeyeceği gözlenmektedir. Arama ve karar alma mekanizmaları ve stratejileriyle ilgili kapsamlı bilgileri ve uygulama alışkanlığı olmayan bilgisayar

mühendisinin varlığını düşünmek bile zordur. Bu açıdan “Yapay zeka” dersinin iki dönemlik (birinin zorunlu olması şartıyla) olması gerekmektedir.

Metodoloji-pedagoji

Metodolojinin genel aracı ders programıdır. Ama ders programı ne kadar iyi olursa olsun iyi uygulama sistemi olmazsa başarı elde edilemez. Bir dönemlik ders süresini ele alalım.

Üniversitelerimizde dönem süresi 14 haftadır (2 haftalık ara sınav süreleri dahil, final sınav süresi hariç). Ara sınav sürelerini çıkartırsak ders süresi 12 hafta olur. Gerçek ders haftalarının sayısı ise 10-11'e düşer (çoğu sayıda bayramları, bayram önu-sonu “tatilleri” dikkata alırsak). Peki, diğer ülkelerde ne kadar? Çok farklı – bazı gelişmiş ülkelerde 16 [5]. Diğer ülkeleri yakalamak için daha fazla çalışmak gerekir, biz daha az çalışırız! Öğrenimin devamlılığının ne kadar önemli olduğunu tüm öğretim üyeleri biliyor. Uzun kesintilerin (dönemiçi 7-10 ders günü, 3 aylık yaz tatili) negatif etkileri çok büyüktür.

Ders programlarının iyi uygulanabilmesinin yollarından biri de uygun metodik ve pedagoji stratejilerin kullanılmasıdır. Bir programlama dilinin öğretilmesi sürecini ele alalım. Geleneksel yöntem şöyledir. Öğretim üyesi bir konuyu anlatır, deyimlerin sintaks ve semantiğini gösterir, basit bir matematiksel problemin algoritmasını oluşturur, kodunu yazar, asistanı dolayısı ile kodun bilgisayarda derlenmesini ve çalışmasını gösterir. Ödev olarak diğer bir basit problemin çözümünü ister. Dönem sonu ödevi olarak da iyi durumda bir kadar karmaşık problemin (çok vakit matematiksel) aynı şekilde çözümü hazırlanır. Bununla dönem bitmiş olur. Öğrencinin burada elde ettiği bilgi birikimi ve programlama alışkanlığı yetersizdir. Onun daha fazla program yazmaya ve orijinal algoritmalar oluşturmaya ihtiyacı vardır.

Şimdi diğer bir yaklaşıma bakalım. Öğretim üyesinin elinde bir proje vardır. Proje için karmaşık bir yazılımın hazırlanması düşünülmektedir. Öğretim üyesi onu birkaç modüllere ayırır, her biri için öğrenci ekipleri oluşturur ve dönem sonu projesi olarak (veya daha uzun süreli) öğrencilere sunar. Böylece öğrenci gerçek bir problemi sıfırdan ele alır, bir yazılım ürününün hazırlanması için gereken tüm etabları geçmiştir.

Dünyanın gelişmiş üniversitelerinde bu yöntem yaygın olarak kullanılmaktadır. Orada bunun yapılması kolay, üniversitelerin yeterince projeleri vardır. Peki türk üniversitelerinde benzer çalışmalar nasıl yapılabılır? Şöyle bir organizasyonu inceleyelim.

Bilgisayar bölümü olan mühendislik fakültelerinde dönem ödevleri ve bitirme çalışmaları fakültenin iki farklı bölüm (örneğin, çevre- bilgisayar) öğrencilerinin ortak çalışmaları biçiminde düşünülebilir. Çevre problemlerinde bol istatistik veriler

vardır veya kolayca elde edilebilir. Ve onlar bu verileri farklı açılardan incelemek için bilgisayar uygulamalarına büyük ihtiyaç duymaktadırlar. Çevre hocası birkaç alt problemden oluşan liste hazırlar, onu bilgisayar hocası ile inceler ve karışık öğrenci ekipleri oluşturulur. Çevre öğrencisi (öğrencileri) problemi araştırır, amacı belirler, bilgisayar öğrencisiyle beraber problemin matematik modelini hazırlar. Bilgisayar öğrencisi gereken algoritmayı oluşturur, programını yazar ve derler. İki öğrenci beraber test verilerini hazırlar ve programı test ederler ve son aşamada yazılımın demosunu yapar ve gereken raporu (raporları) hazırlarlar. Sonuçta öğrenciler kendi alanında yeterli bilgi sahibi olur, disiplinlerarası çalışmanın ne olduğunu anlar, ekip çalışması ile ilgili gerçek deneyim elde ederler. Yerel veri tabanlarının oluşturulması, verilerin düzenlenmesi ve genelleştirilmesi, belli özellikli bilgilerin genel veri topluluğundan çıkartılması gibi problemler ortaklaşa çalışma konuları olabilir. Diğer bölümlerle de (kimya mühendisliği-bilgisayar, malzeme mühendisliği-bilgisayar vs) aynı şekilde çalışmaların organize edilmesi faydalı olur. Bilgisayar eğitiminin iyileştirilmesini engelleyen nedenlerden biri de bilgisayar dalında türkçe belli bir terimler sözlüğünün olmamasıdır. Biz burada türkçe veya ingilizce eğitim problemlerine girmeyeceğiz. Ama bilgisayar gibi karmaşık bir bilim dalının disiplinlerinin ana dilde daha kolay ve verimli öğretilebileceği gerçeğini de unutmayalım. Sözü geçen sözlüğün hazırlanması ve üniversitelerde zorunlu olarak kullanılması acil çözülmesi gereken problemlerden bir tanesidir.

Eğitimin kalitesi

Bugünlerde Avrupa’da üniversite eğitiminin kalitesinin yükseltilmesi ile ilgili konular çok konuşuluyor. Türkiye’de de çok konuşuluyor ve konuşulmuştur. Bizim işimiz daha zor, önce avrupalıları yakalamak gerekir. Kaliteyi söz koşturmakla yükseltmek olmaz. Önce “Yükseltme kriterileri “ hazırlamak, sonra ise onlara dayanarak uygulama mekanizması geliştirmek gerekmektedir. Varmı elimizde böyle bir kriteriler seti? Belki var. Olmadığını varsayarak onun oluşturulmasına katkı sağlayacak bazı soruları ele alalım. İngiliz tiyatro yazarı Bernard Shaw’nun bir deyimini hatırlayalım: “Those who can, do, those who can’t, teach.” Yorum ilgiliye kalır. Dolaylı yorum yaparak bunu “Öğretmek yapmaktan zordur” gibi anlayalım. Bu yorum uzman yazılımcılar arasında yaygın olan “Öğretmek program geliştirmekten zordur” fikiri ile de örtüşmektedir. Yani öğretmek çok zor olan bir iştir. Burada yükümlü bilgi birikiminin yanısıra büyük deneyimin olması da gerekmektedir. Gelişmiş ülkelerde çoğunlukta derse (lecture) yalnız profesörler ve doçentler girebilir. Diğerleri giremezler, asistanlık yaparlar, tecrübe toplamak

lazım. Yeni mazun olmuş (veya doktorasını yeni bitirmiş) çünkü öğrenci bugün arkadaşlarının karşısına çıkarak halen kendinin tam olarak anlamadığı bir konuyu anlatmaya çalışırsa şu öğretimin kaliteli olacağı düşünülemez. Evet, öğretim üyesi yetmiyor. Peki, kalite ne olsun? Deneyimi az olan veya hiç olmayan bir doktorun amaliyat etmeğe kalkıştığını gördünüz mü? Ama bir tane bile orta seviyeli program yazıp derlemeyen öğretim görevlisi kolayca programlamayı “öğretebilir”.

Araştırma-egitim ilişkilerine göz atalım. Akademik yükseltme için bugün kullanılam temel gösterge uluslararası yayınların olmasıdır. Çok zaman bu yayınların içeriği değil sadece sayısı göz önüne alınır. Yayınların içeriğini, gerçek bir araştırmannın ürünü olup-olmadığını, potansiyel faydalılığını şimdilik göz ardı edelim. Diğer soruya bakalım: neden adayın eğitimcilik özellikleri önemli sayılmıyor? Doçent ve profesör ünvanlarının eğitim ünvanları olduğu neden unutuluyor? Öğretmeğin araştırmadan daha zor bir iş olduğunu bir daha hatırlayalım.

İyi araştırmacı her zaman iyi öğretmen değildir. Terside doğrudur. Tabi, öğretim üyesinde her iki özelliğin olması istenilen durumdur. Maalesef çok zaman böyle olmuyor. Peki, diğer ülkelerde bu probleme bakış açısı nasıldır? Diğerlerinin tecrübesini öğrenmekte her zaman fayda vardır. Bununla ilgili bir olayı hatırlamak yerinde olur. 2003 yılıydı. Medyada Türkiye’de teknolojinin geliştirilmesinin hızlandırılması, yeni teknolojilerin, özellikle bilgisayar teknolojilerinin ülkeye getirilmesi, gereken uzmanların kısa sürede yetiştirilmesiyle ilgili bir kampanya başlatılmıştı. Başbakan bir toplantıda bu konuda her tür öneriyi açık olduklarını, gereken yardımı vereceklerini söylemişti. Bu zamana denk gelen bir bilgisayar bölüm başkanları toplantısında şu yazının yazarı şöyle bir öneride bulundu. Toplantı bir heyet oluşturup uygun devlet kurumları dolayısıyla Hindistana göndersin. Amaç, bu ülkenin son yıllarda yazılım eğitimi ve teknolojilerinde elde ettiği başarılarla tanışlık, üniversiteler arasında gereken ilişkilerin kurulması. Önerinin faydalı olacağı, en azından tartışılacağı beklenirdi. Ama kimseden ses çıkmadı. Tek bir katılımcı (bölüm başkanı değildi) negatif yaklaşım sergiledi, bu işin faydasının olmayacağını, bizim onlardan öğreneceğimiz bir şeyin olmadığını söyledi. Yanıbaşınızda ortamı sizinkine benzer bir komşunuz büyük başarılar elde etmiş, ama siz bununla ilgilenmiyorsunuz. Bölüm başkanları genelde çağdaş ve akıllı insanlardır, böyle bir fikiri kendileri de düşünmüş olabilir. Ama somut bir adım atılmadı.

Sonraki gelişmeler önerinin önemini kanıtladı. İki yıl sonra (2005) gazetelerde bir yazı geçti. Bahçeşehir üniversitesinden bir heyet Bangalor kentinde (Hindistan) ziyarette bulunmuş, ziyaretin

amaçlarından biri yazılım mühendisliği ve eğitimiyle ilgiliymiş. Böyle bir ziyaret daha müsait zamanda ve bilgisayar bölüm başkanları toplantısı gibi önemli bir organizasyon tarafından yapılsaydı daha iyi olmazdı mı?

Son söz

Son yıllarda Avrupa’da üniversite eğitiminin kalitesinin yetersiz olduğu, onun yükseltilmesinin gerekliliği problemleri gündemdedir. Ülkemiz üniversiteleri için de bu problemler önemlidir. Bu yazıda bu problemlerin birkaçı ele alınmış, “kalitenin yükseltilebilmesi” için önce bir kriterler setinin oluşturulmasının gerekliliği gösterilmiş, tartışılması gereken öneriler sunulmuştur.

Kaynakça

1. Wankut, P.C. & Oreovicz, F.S., Teaching Engineering, Purdue University, <http://engineering.purdue.edu/ChE/News>

- [and_Events/Publications/teaching_engineering/](#)
2. Kennedy, L.A., The Educational Process, Global Journal of Engineering Education, Vol. 3, No. 1, 1999
3. ACM/IEEE-Curriculum 2001 Task Force, Computing Curricula, software Engineering 2004, June 2004, <http://sites.computer.org/ccse/>
4. Hilburn, T.B. & Towhidnejad M., How does Software Engineering fit into a College of Engineering?, Proceedings of the 2005 American Society for Engineering Education Annual Conference & Expositions, 2005.
5. Kriviskas, R.V., Active Learning at Kaunas University of Technology, Global Journal of Engineering Education, Vol. 9, No. 1, 2005.