

Nesne Tabanlı Programlama

Bölüm 1

Giriş

Dr. Öğr. Üyesi Murat TAŞYÜREK (kayubmprogramlama1@gmail.com)

2 Ekim 2023

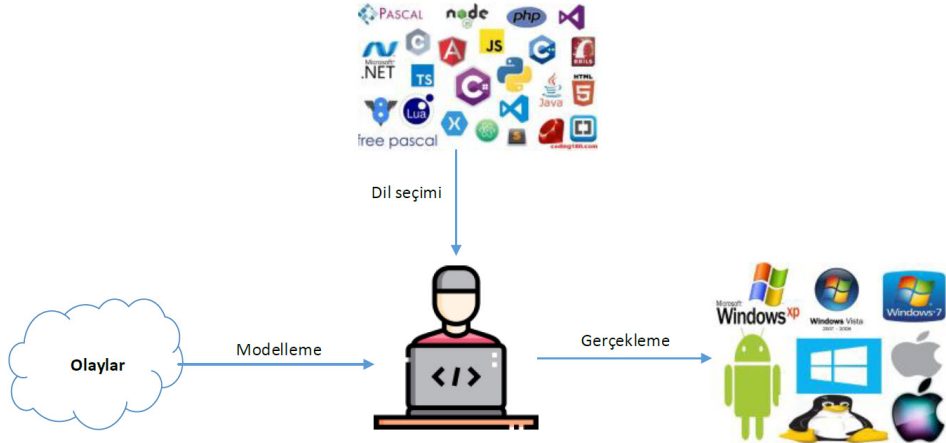
Kayseri Üniversitesi, Bilgisayar Mühendisliği Bölümü

- Programlama dili olarak C# kullanacağız.
- Editör olarak Visual Studio kullanacağız.
- Projelerden (ödevlerden) 40 puan alacaksınız, vize-final 60 puan üzerinden olacak.

- Daha önceden C# bilmeyen ve Visual Studio kullanmayan arkadaşlar için
- Bilgilerini tekrar etmek isteyen arkadaşlar için
- **İndirip uygulayınız**

- Konuşma dillerinin kurallarına benzer şekilde programlama dillerinin de belirli kuralları vardır.
- Syntaks bilginizin iyi olması iyi bir yazılım geliştirmek için yeterli değildir.
- Program veya diğer bir ifade ile yazılım, gerçek hayatta karşılaşılan olayların çözümünün bilgisayar sistemlerinde ifade edilmesidir.
- İlgili yazılımı geliştirmek için öncelikle tasarım aşamasında olayı oluşturan varlıkların bilgisayar sistemlerinin anlayacağı şekilde modellerinin oluşturulması gerekmektedir.

Model Oluşturma



Kullanıcı açısından bir yazılımın kalitesi

- Geliştirilen yazılım istenen işi doğru olarak yapmalıdır.
- Kullanıcı dostu arayüze sahip olmalıdır.
- Geliştirilen yazılım hızlı çalışmalıdır.
- İşlemci, ram ve disk gibi sistem kaynaklarını az kullanmalıdır.
- Yazılım sağlam (stable) çalışmalıdır.
- Yazılım eğitim dökümanına sahip olmalıdır.

Geliştirici açısından bir yazılımın kalitesi

- İlgili yazılımın kaynak kodu anlaşılabilir olmalıdır.
- Modüler bir yapıda olmalıdır.
- Modüller tekrar kullanılabilir bir yapıda olmalıdır.
- Bakımı kolay olmalıdır.
- Geliştirme maliyeti düşük olmalıdır.
- İlgili yazılım tasarım aşamalarını açıklayan dökümantasyona sahip olmalıdır.

ISO Standartlarına göre bir yazılımın kalitesi

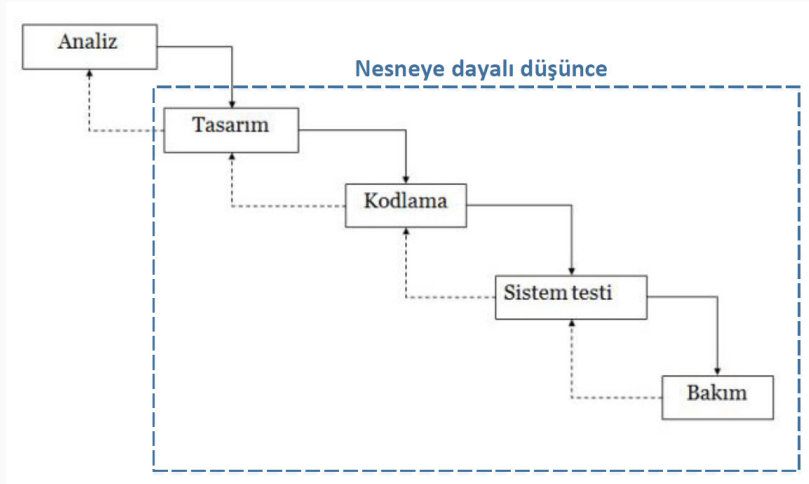


ISO-9126 Üst Seviyeli 6 Kalite Faktörü

- ISO 9126 yazılımın değerlendirilmesi için uluslararası bir standarttır.
- **İşlevsellik;** Yazılım ürününün belirlenmiş koşullar altında kullanıldığında belirtilen ve istenilen ihtiyaçları ve işlevini yerine getirme kabiliyetidir.
- **Güvenilirlik;** Yazılım ürününün belirlenmiş koşullar altında kullanıldığında belirli bir performans seviyesini sürdürme kabiliyetidir.
- **Kullanılabilirlik;** Yazılım ürününün belirlenmiş koşullar altında kullanıldığında kullanıcı tarafından anlaşılması, öğrenilmesi, kullanılması ve kullanıcıya çekici gelme kabiliyetidir.
- **Verimlilik;** Yazılım ürününün belirlenmiş koşullar altında kullanıldığında kullanılan kaynak miktarına uygun performans sağlamasıdır.
- **Sürdürülebilirlik;** Yazılım ürününün değişiklik yapılabilme kabiliyetidir. Değişiklikler düzeltme iyileştirmeleri veya çevredeki yazılım değişikliklerinin, gereksinimlerinin ve fonksiyonel özelleştirmelerin adaptasyonunu içerebilir.
- **Taşınabilirlik;** Yazılım ürününün bir çevreden başka bir çevreye aktarılma kabiliyetidir.

Neden nesne tabanlı programlama

- Nesne tabanlı programlama önceki slaytlarda sunulan gereksinimleri sağlamak üzere ortaya konmuş bir yöntemdir.
- Tasarım yapılırken ve bu tasarım koda dönüştürülürken yazılımın kalitesini gösteren maddeler dikkate alınmalıdır.

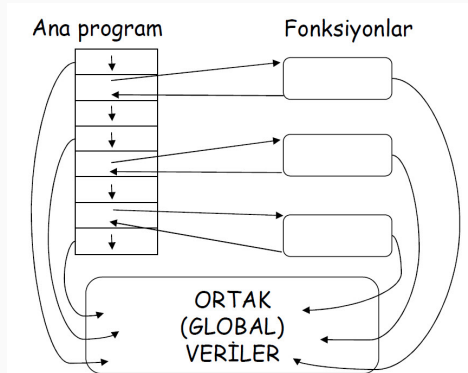


Yazılım geliştirme araçları

- **Analiz:** Yazılımın ne yapacağı bu aşamada belirlenir. Yazılacak programın hedefleri saptanır.
- **Tasarım :** Yazılımın nasıl yapılacağına karar verilir. İki aşamadan oluşur. Programın mantıksal tasarımı yapılır. Sonrasında ayrıntılı tasarımı gerçekleştirilir.
- **Kodlama:** Programın yazılacağı programlama dili belirlenir. Program yazılır. Kodlama sürecinde yazılan programın istenileni karşılayıp karşılamadığı sürekli test edilir.
- **Test:** Elde edilen yazılımın sına ve doğrulama aşamasıdır. Yazılımın belirlenen gereksinimleri sağlayıp sağlamadığı ve gerçekleştirimin beklentileri karşılayıp karşılamadığı test edilir.
- **Bakım:** Yazılım kullanılmaya başlandıktan sonraki yazılımın desteklenmesi aşamasıdır. Yazılımdaki eksikliklerin ve hataların giderilmesi, performans iyileştirmesi ve benzer alt aşamaları içerir.

İşlev (procedure) tabanlı programlama yöntemi

- Bu programlama yönteminde öncelikle yapılması gereken iş belirlenir. Basic, Fortran, Pascal ve C gibi programlama dilleri tarafından desteklenmektedir.
- Büyük boyutlu ve karmaşık işler, daha küçük ve daha basit işlere (fonksiyon) bölünerek gerçekleştirilir.



İşlev tabanlı programlama yöntemini inceleme

- Böl ve yönet prensibine dayanır. Amaç büyük programları küçük parçalara bölerek yazılım geliştirme işini kolaylaştırmaktadır.
- Ancak, gerçek dünyadaki sistemler sadece fonksiyonlardan oluşmaz.
- Fonskiyona ağırlık verildiğinden veriler göz ardı edilir ve bu da veride bozulmalara sebep olur.
- Yazılım geliştirici kendi veri türünü oluşturamaz.
- Güncellemek ve yeni öğeler eklemek zordur.
- İşlev tabanlı programlama yöntemi kullanılarak da kaliteli yazılımlar geliştirilebilir.
- Fakat, nesne tabanlı programlama kaliteli bir yazılımın oluşturulması için daha çok olanaklara sahiptir.

Nesne tabanlı programlama (object oriented programming) yöntemi

- Gerçek dünya nesnelerden oluşmaktadır. Bu bakış açısıyla nesneye yönelik geliştirilecek bir yazılımda çözülmek istenen problemi oluşturan nesneler gerçek dünyadaki yapılarına uyumlu bir şekilde bilgisayarda modellenir.
- Her nesne belli bir sorumluluğa göre tasarlanır.
- Nesneler nitelik (özellik, feature, property) ve davranış (yetenek, fonksiyon, procedure) olmak üzere iki bölümden oluşur.

Sınıfların belirlenmesi

- Nesne tabanlı bir programda öncelikle olaya ait nesneler belirlenmelidir.
- Örneğin insan kaynakları ile ilgili bir program yazıyorsanız nesneler memur, işçi, müdür ve genel müdür şeklinde olur.
- Nesneye-yönelik tasarımda ilk adım programın ihtiyaç duyacağı sınıfların belirlenmesidir.
- Memur, işçi, müdür ve genel müdürün ortak özelliği insan olmasıdır. Bu nedenle insan türünde bir sınıf oluşturulmalı ve diğer nesneler bu sınıftan türetilmelidir.
- Geometrik bir program yazdığımızda nesneler üçgen, dörtgen, kare vb şekilde olacaktır. Bu nesnelerin ortak özellikleri ise hepsinin nokta içermesidir. Nokta türünde bir sınıf tanımlanmalı ve bu nesneler o sınıftan veya o sınıfın türevlerinden üretilmelidir.

Örnek nokta nesnesi

```
using System;

namespace Örnek1
{
    0 references
    class Program
    {
        0 references
        static void Main(string[] args)
        {
            Console.WriteLine("Hello World!");
            Nokta nn = new Nokta(); //nense ←
            nn.HareketEttir(50, 50);
        }
    }
    2 references
    public class Nokta //sınıf ←
    {
        int x, y; //özellik ←
        1 reference
        public void HareketEttir(int yeni_x, int yeni_y) //yetenek ←
        {
            this.x = yeni_x;
            this.y = yeni_y;
        }
    }
}
```

Nesne tabanlı yöntemin değerlendirilmesi

- Gerçek dünya nesnelerinden oluştuğundan dolayı daha gerçekçi modeller oluşturulacaktır. Program daha anlaşılır olur.
- Nesne modelinde veriler sadece üye fonksiyonların erişebileceği şekilde düzenlenebilir. Diğer bir ifade ile veri saklama (data hiding) yapılabilir.
- Yazılım geliştiriciler kendi veri türlerini üretebilir.
- Bir model oluşturulduktan sonra bu model farklı şekillerde defalarca kullanılabilir (reusability).
- Programı güncellemek daha kolaydır.
- Nesne tabanlı programlama yöntemi takım çalışması için uygundur.