

İTÜ Bilgisayar ve Bilişim Fakültesi
Bilgisayar Mühendisliği Bölümü

Proje Planı

Kötü Kodların Belirlenmesi

Detecting bad codes in object-oriented programs

Murat Sekin
040040243

09.10.2015

1. Projenin Amacı

Nesneye yönelik programlama dili kullanılarak yazılmış kaynak kodlarda kötü kodları tespit edecek bir program geliştirilecektir. İyi kod yazımının özendirilmesi amaçlanmaktadır.

Yazılım kalitesini artırmak için çeşitli standartlar geliştirilmiştir. Bugün iyi olarak kabul edilen kodlama teknikleri de mevcuttur. Bunlara uyulmaması kötü kod yazılmasına neden olabilir. Kötü yazılmış bir kodun okunması, anlaşılması ve bakımı zordur. Programların performansını etkileyebilir ve ilerde hata çıkmasına neden olabilir. Kodların derlenmesini ve çalışmasını engellemez.

Kötü kodların nedenleri:

- Analiz ve tasarımın yeterli ve doğru olmaması
- Yazım kurallarına ve isimlendirme kurallarına uyulmaması
- Tutarsızlık
- Biçime ve formata dikkat edilmemesi
- Tekrar eden kod parçaları
- Yorum olmaması

Kötü kod örnekleri :

```
// goto kullanmak. program akışını takip etmeyi zorlaştırır
goto label;
...
label: statement;

// büyük boyutlu vektör fonksiyondan geri döndürmek
std::vector<std::string> BuildLargeVector();
...
std::vector<std::string> v = BuildLargeVector();

// magic number kullanmak
for(int i=0; i < 10; i++)
{
    ...
}
```

2. Projenin Kapsamı

Hedef dil olarak C++ programlama dili seçilmiştir. Bu programlama dili kullanılarak yazılmış programlardaki kötü kodlar bulunacak ve kullanıcıya gösterilecek.

Masaüstü uygulaması olacak.

C# programlama dili ile geliştirilecek.

.NET platformu üzerinde çalışacak.

C++ dilinin parse edilmesi için 3. parti bir yazılım kullanılacak.

3. Zamanlama

Projenin iki önemli aşaması bulunmaktadır. İlk olarak kaynak kodların parse edilmesi gerekmektedir. C++'ın karmaşıklığı nedeniyle bu iş için 3. parti yazılım kullanılacak. İkinci aşama ise parserin çıktılarının anlamlandırılmasıdır.

Task Mode ▼	Task Name ▼	Duration ▼	Start ▼	Finish ▼	Predecessors ▼
→	C++ öğrenilmesi	10 days	12.10.2015	23.10.2015	
→	Parser araştırılması	5 days	26.10.2015	30.10.2015	1
→	Kötü kod örneklerinin araştırılması	5 days	2.11.2015	6.11.2015	2
→	Tasarım	5 days	9.11.2015	13.11.2015	3
→	Kodlama	110 days	16.11.2015	15.4.2016	4
→	Ara rapor yazımı	10 days	1.2.2016	12.2.2016	
→	Test	10 days	18.4.2016	29.4.2016	5
→	Rapor yazımı	20 days	2.5.2016	27.5.2016	7

Zaman kalması halinde performans, güvenlik vb. ile ilgili kötü kodların tespit edilmesi için çalışma yapılacaktır.