

# PROGRAMLAMA LABORATUVARI 2

## 2. PROJE

Muhammet Ali BAKINÇ  
Abdulhalik SARIÇİÇEK  
Bilgisayar Mühendisliği Bölümü  
Kocaeli Üniversitesi

### Özet

Bu doküman Programlama Laboratuvarı 2 dersi 2. Projesi için çözümümüzü açıklamaya yönelik oluşturulmuştur. Dokümanda projenin tanımı, çözüme yönelik yapılan araştırmalar, kullanılan yöntemler, proje hazırlanırken kullanılan geliştirme ortamı ve kod bilgisi gibi programın oluşumunu açıklayan başlıklara yer verilmiştir. Doküman sonunda projemizi hazırlarken kullandığımız kaynaklar ve proje derlenirken dikkat edilmesi gereken hususlar bulunmaktadır.

### 1. Proje Tanımı

Projede bizden istenen, bir otomatik araç yıkama makinesinin minimum sayıda para üstü vererek çalışmasıdır. Bunun için sanal ya da fiziksel geliştirme kartları da kullanılabilir.

Tasarlayacağımız makinenin, kasa bilgilerinin metin belgesinden okumasını, müşteriden para girişi ve hizmet seçimi yapmasını istemesi ve varsa para üstünü en az sayıda banknot kullanarak vermesi gerekmektedir. Önceden belirlenen şablona göre kasa bilgileri oluşturulacak, makine kasadaki para ve malzeme bilgilerine göre işlemlerini yapacaktır. Makinede para sıkışması gibi durumlar kontrol edilecektir. Kasada bir banknotun olmaması halinde bir diğeriyle hesaplama yapılacaktır, para olmaması halinde işlemler iptal edilip para iade edilecektir.

Program işleyişi aşağıdaki gibi olmalıdır:

- Program metin belgesinden kasadaki para miktarlarını ve malzeme bilgilerini okur.
- Müşteriden para girişi yapmasını bekler. Müşteri tamamlayana kadar işlemi tekrarlar.
- Müşteriden istediği hizmetleri seçmesini bekler. Müşteri birden fazla işlem seçebilir.
- Para sıkışma durumunu kontrol eder. Sorun varsa işlemleri iptal eder ve parayı iade eder.
- Sorun yoksa para üstünü kasadaki bilgilere göre hesaplar ve müşteriye verir.

### 2. Araştırmalar ve Yöntem

Projeye, proje tanımında anlatılan metin belgesini oluşturmakla başladık.

Projede nasıl yapılması gerektiğini düşündüren en önemli nokta belge okuma ve yazma işlemini çözmektir. Bunun için çeşitli kaynaklardan kodlar inceledik. Belgeyi doğru okuyabilmek ve yeniden yazabilmek en temel problemdir. Bunu doğru şekilde yapabildikten sonra geriye kalan kısımları daha rahat yapabildik. (Geliştirme kartı kullanımı için ekstra araştırmalar yapılması gerekmektedir.)

### 3. Geliştirme Ortamı

Projemizi Windows sistemde, *Dev-C++* üzerinde geliştirip derledik. *C* dilini kullandık.

## 4. Kod Bilgisi

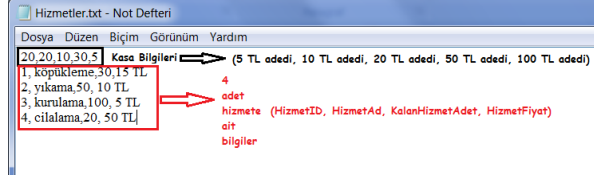
### 4.1. UML Diyagramı ve Akış Şeması

Son sayfadadır.

### 4.2. Algoritma

Bu kısımda projenin genel algoritmasına açıklık getireceğiz.

Metin belgesinin belirlenen şekli şöyledir:



Program başladığında metin belgesi okunur. Bunun için *dosyaOku* fonksiyonu çağrılır. İlk satırda sırasıyla 5, 10, 20, 50, 100 TL'lik banknot miktarları virgüller ile ayrılmıştır. Bunlar okunur ve boyutu 5 olan *para* dizisine sırasıyla atılır. Diğer satırlarda ise hizmetler bulunmaktadır. Sırasıyla *id*, *hizmet adı*, *malzeme miktarı* ve *fiyat bilgisi* vardır. Hizmet isimleri ayrı ayrı değişkenlerde tutulur. Malzeme miktarları ve fiyat bilgileri de boyutu 4 olan *hizmetMiktar* ve *hizmetFiyat* dizilerine sırasıyla atılır.

Daha sonra *paraYukle* fonksiyonu çağrılır. Bu fonksiyonda müşteriden para girişi yapması istenmektedir. Sırasıyla 5, 10, 20, 50 ve 100 TL için A, B, C, D, E, çıkış için X karakterleri belirlenmiştir (Buton da kullanılabilir). Girilen karakterlere göre *yatirilan* isimli değişkene miktarlar eklenir. Para dizisindeki banknot miktarları da arttırılır. Tamamlandıktan sonra *hizmetSec* fonksiyonu çağrılır.

Yine para yüklemede olduğu gibi müşteriden hizmet seçmesi beklenir. Sırasıyla köpükleme, yıkama, kurutma ve cilalama için A, B, C, D, çıkış için X karakteri beklenir. Seçilenler için *hizmetler* isimli dizide yine sırayla kaç adet seçildikleri saklanır. Seçim yapılırken o hizmetin malzeme miktarının olması ve yatırılan paranın yetmesi durumları kontrol edilir. İşlem tamamlandıktan sonra 1-4 arası rastgele bir sayı seçilir. Sayı 2 ise para sıkışma durumu, değilse normal durum geçerli olur.

Eğer para sıkışma belirlendiyse *paraSikisma* fonksiyonu çağrılır. Yatırılan tutar yine en az sayıda para ile iade edilir. Bunun için sırasıyla 100 liradan 5 liraya kadar bölünerek en az miktar bulunmaya çalışılır. Aynı zamanda *para* dizisindeki banknot miktarları da kontrol edilir.

Normal durumda ise *paraUstu* fonksiyonu çağrılır. Harcanan miktar alınan hizmetler ve fiyatlarına göre hesaplanır. Kalan miktara para sıkışma durumunda yapılan hesaplama ile en az banknot hesaplanır.

İki durum sonunda da *kasaGuncelle* fonksiyonu ile güncel miktarlar aynı şablona uygun şekilde metin belgesine yazılır. Program tamamlanır.

### 4.3. İstatistik

Program kodu boşluklar dahil yaklaşık 350 satırdan oluşmaktadır. Kullandığımız kütüphaneler ve ne için kullandığımız kabaca aşağıdaki gibidir:

*stdlib.h*

Random sayı belirlemede kullanıldı.

*stdio.h* - *conio.h*

Temel fonksiyonlar ve dosya okuma işlemleri için kullanıldı.

*string.h*

Stringe metin ekleme ve kopyalama işlemlerinde kullanıldı.

### 4.4. Programın Derlenmesi

Program kaynak kodu ve 1 metin belgesinden oluşmaktadır. Bu projeyi *CodeBlocks*, *Dev-C++* veya farklı bir C IDE'si kullanarak derleyebilirsiniz. Derlerken dikkat edilmesi gereken mesele ise kaynak klasöre metin belgesinin eklenmiş olması gereklidir.

### Kaynakça

1. C fonksiyonları için,

[https://www.tutorialspoint.com/cprogramming/c\\_functions.htm/](https://www.tutorialspoint.com/cprogramming/c_functions.htm/)

2. Dosya okuma işlemleri için,

<https://www.programiz.com/c-programming/examples/read-file>

3. Çeşitli sorunları çözmek için,

<https://stackoverflow.com/>

