**BESTE TURKMEN -TCP SOCKET PROGRAMLAMA**

**-ATM OTOMASYONU PROJESİ**

**ÇALIŞMA SAYFASI**

**DOSYA YAZDIRMA**

C++ Programlama Diliyle Dosya İşlemleri

* Bu işlemler için fstream kütüphanesinde yer alan ifstream ve ofstream sınıfları kullanılır.
* Dosya oluşturmak için ofstream sınıfının kurucu methoduna dosya ismini format belirterek yazmak yeterlidir.
* Oluşturulan dosyaya yazmak için ise yazılacak değerin ofstream değişkenine yönlendirilmesi yeterlidir.
* **Is\_open()** metodu ile dosyanın açılır olup olmadığının kontrolu yapılabilir.
* Dosya okuma işlemi için **ifstream** sınıfının kurucu metoduna dosya adının verilmesi ve okuma biçimine göre **getline** veya **get** metodunun kullanılması yeterli olacaktır.
* Dosya silme işlemi için **remove** fonksiyonu kullanılır.
* Dosyanın sonuna ekleme yapmak için **ofstream** sınıfına ait kurucu metodun ikinci parametresine **ios::app** değerinin yazılması yeterli olacaktır.

**SOCKET PROGRAMLAMA**

\*Soket programlama bilgisayarlar arası veri haberleşmesinde kullanılan,

temel olarak alıcı ve gönderici arasındaki iletişim yönetmek için kullanılan

programlama tekniğine denir.

\*HTTP, FTP, SMTP gibi iletişim protokolleri soket yapısı üzerine inşa edilmiştir.

\*Soket programlama ile kendi kurallarımızı belirleyerek chat-sohbet,

dosya transferi gibi uygulamalar yapabiliriz.(Dosya Transferi araştır)

**yazdırma ve okuma işlemleri için**

-stdin - veri girişi

-stdout- veri yazdırma

-stderr- hata işleme

fopen() fonksiyonu. Filename parametresi ile gösterilen dosyayı mode parametresi

ile gösterilen dosya erişim modunda açar ve bu dosyaya ait FILE dosya akışına

bir işaretçi geri döndürür. Açılan dosya ile ilgili tüm işlemler bu fonksiyon

tarafından geri döndürülen FILE işaretçisi ile yapılır.

\*Windows işletim sistemlerinde Winsock arayüzü kullanılır.

#include <winsock2.h>

#include <ws2tcpip.h>

kütüphaneleri eklenmelidir

\*İki bilgisayar arası haberleşme için her iki bilgisayarında soket yolunu socket

fonksiyonu ile açması gerekir.

**int socket(int domain, int type, int protocol);**

Fonksiyonun domain parametresi kullanılacak olan protokol türünü, type kullanılacak

veri gönderme protokolünü, protocol parametresi ise kullanılacak taşıma protokolünü belirtir.

Domain parametresi **AF\_INET (IPv4)**

Type parametresi ise **SOCK\_STREAM (TCP)**

Protocol parametresi de genellikle **0** değerini verilir.

**SUNUCUNUN OLUŞTURULMASI (SERVER)**

int bind(int sockfd, const struct sockaddr \*addr, socklen\_t addrlen);

Fonksiyon soketi belirlenen IP adresi ve port ile ilişkilendirir.

Fonksiyonun ilk parametresi **socket** fonksiyonu ile oluşturulan değeri, ikinci parametresi adres bilgilerini, son parametre ise adres bilgisi uzunluğunu alır.

Adres bilgisi kullanılan IP yapılandırmasına göre IPV4 (AF\_INET) için **sockaddr\_in** veri yapısını ile oluşturulur.

Ayarlar oluşturulduktan sonra **sockaddr** türüne dönüşüm yapılır.

İşlemciler verileri bellekte little endian ve big endian türüne göre tuttuğundan IP ayarlarının oluşturulması sırasında htonl, htons, ntohl, ntohs fonksiyonlarının kullanılması olası hataları en aza indirecektir.

Soket IP adresi ve port ile ilişkilendirildikten sonra istekleri dinlemek için **listen** fonksiyonu kullanılır.

Dinleme işleminden sonra artık sunucu hazır hale gelmiş olur

Sunucu bu işlemlerden sonra gelen istekleri accept fonksiyonu ile kabul etmesi gerekir.

**int accept(int sockfd, struct sockaddr \*addr , socklen\_t \*addrlen);**

**BANKAMATİK OTOMASYONU**

**Programı başlat**

**Ana sayfa**

1)YETKİLİ GİRİŞİ

2)MÜŞTER İ GİRİŞİ

3PROGRAMI KAPAT

**YETKİLİ GİRİŞİ**

ADMİN: PITON

ŞİFRE :123

1)MUSTERİ EKLE

2)MUSTERİ SİL

3)ÇIKIŞ

**MUSTERİ EKLE**

TC : cout<<”tc :”; cin>>tc;

AD-SOYAD:

DOĞUM TARİHİ :

CİNSİYET:

ADRES:

TEL:

E POSTA:

Bilgilerini gir:

DOSYA OLUSTUR VE BİLGİLERİ ORAYA AKTAR

**MUSTERİ SİL:**

TC numarası ile karşılaştırma yap

TC YANLIŞSA BİLGİLERİ DOSYADA SİL

**MUSTERİ GİRİSİ**

**AD-SOYAD:**

**NUMARA:**

1)PARA CEK

2)PARA YATI

3)HAVALE/EFT

4)BAKIYE SORGULA

5)SIFRE DEĞİŞTİR

6HAREKET DOKUMU

0)CIKIS