Bilgisayar Haberleşmesi

Ödev II Soket Api Kullanılarak Veri Aktarımı

Beycan Kahraman 040020337 beycan@mail.com

Doç. Dr. Sema Oktuğ Ar. Gör. Tacettin Ayar 04.12.2006 Verilen ödevi 3 farklı işletim sisteminde çalıştırıp denedim. Bunlar *Microsoft Windows XP SP II, Pardus Linux 2007 Beta* ve *Kubuntu 6.06.1 LTS Dapper Stable* sistemleridir.

İstenen programı *Microsoft Windows Visual C++ 6.0* ortamında hazırladım. Hazırlanan kodun çalışması için Visual C++'ta

Project -> Settings -> Link -> Object/Library Modules *Ws2_32.lib* kütüphanesinin eklenmesi gerekmektedir.

Linux'lerde programı çalıştırmak için ise iki adet konsol açılıp aşağıdaki derlemeler ve çalıştırmalar yapılmıştır.

```
deejay@adelante:~$ gcc 040020337_s -o sunucu
deejay@adelante:~$ ./sunucu

deejay@adelante:~$ gcc 040020337_c -o istemci
deejay@adelante:~$ ./istemci
```

Programa eklenen kısımlar aşağıda verilmiştir.

Sunucu program 8 bitlik veriyi aldıktan sonra eşlik bitini kontrol ederek, sonuca göre istemciye ACK veya NACK mesajı gönderir. Ayrıca, veriyi dosyaya yazıp yazmayacağına karar verir.

```
f = fopen(DOSYA, "w");
                                         // open file
while(fileEnd)
     n = recv(sd2, buf, 8, 0);
                                         // get bits
     if(n < 8) fileEnd = 0;
                                         // is file end?
     buf[n] = ' \setminus 0';
     totalOnes = 0;
     for(i=0; i<n; ++i)</pre>
          if(buf[i] == '1')
               ++totalOnes;
                                         // calculate eslik bit
     if(totalOnes%2)
     {
          printf("%s << err\n"):</pre>
          send(sd2, "1001", 4, 0);
                                         // NACK
     }
     else
          if(n > 0)
          buf[n-1] = '\0';  // NO ERROR fprintf(f, "%s", buf);  // write to file send(sd2, "0110", 4, 0);  // ACK
     }
}
fclose(f);
                                         // close file
```

İstemci program ise dosyadan okuduğu her 7 bitlik kısım için eşlik biti üreterek veriyi karşıya gönderir. Hata olma olasılıklarını eklemek için, rasgele oluşturulan hatalar sonucunda gelen ACK veya NACK verisine göre, eski veri tekrar gönderilebilir.

```
// 4 bit ACK
err[4] = '\0';
srand(time(NULL));
f = fopen(DOSYA, "r");
                                 // open file
while(fileEnd)
     if(noError)
                                 // send again ?
         fscanf(f, "%c", &buf[i]);
              if(feof(f))
                   fileEnd = 0;
                   buf[i+1] = ' \setminus 0';
                   break;
              }
         }
         buf[i+1] = ' \setminus 0';
         totalOnes = 0;
         for(n=0; n<i; ++n)
              if(buf[n] == '1')
                   ++totalOnes; // calculate eslik bit
         buf[i] = totalOnes%2 == 1 ? '1' : '0';  // write eslik bit
     }
     if(rand()%4002 < 0337 \&\& i > 3) // let's create an error
         strcpy(buffErr, buf);
         n = rand()\%(i+1);
                                      // on which bit
         buffErr[n] = buffErr[n] == '1' ? '0' : '1';
         send(sd, buffErr, i+1, 0);
     }
     else
         send(sd, buf, i+1, 0);
                                      // no error
     n = recv(sd, err, 4, 0);
                                      // get ACK
     noError = 1;
     if(err[0] == '1' || err[3] == '1')
         noError = 0:
                                       // NACK
}
fclose(f);
                                       // close file
```

Sonuç olarak, değişik işletim sistemlerinde ortak çalışabilen bir yapı oluşturulması programcı için oldukça yararlı oluyor.