

**Bilgisayar Ağları dersi**

**Doç. Dr. Zafer ALBAYRAK**

**Arş. Gör. Muhammed Yusuf KÜÇÜKKARA**

**Teknoloji Fakültesi**

**Bilgisayar Mühendisliği 2.sınıf**

**OgrenciNo = “B210109048”**

**AdSoyad = “Furkan Duran”**

**2.ÖDEV Kablolu 25 Düğümlü Ağın Test Edilmesi, Simülasyonunun Gerçekleştirilmesi**

Bu ödevimizde NS2 uygulaması üzerinden 25 düğümlü oluşturmuş olduğum ağın test edilmesini ve simülasyonun gerçekleştirilmesine değineceğim.

Öncelikle ağımızı oluşturmamız için bir topoloji kullanmalıyız. Ben ağın yönetimi kolay olması açısından star topolojisini kullandım. Star topolojisinde bilgisayarlar merkezdeki bir hub ya da switch üzerinden birbirleri ile bağlantı kurarak haberleşme sağlanır. Star topolojisinin diğer yararı ise hergangi bir düğümde çıkan aksaklıkta diğer düğümlerin etkilenmeyip sorunsuz çalışması. Dezavantajı olarak da hubdaki sıkıntı sistemin çalışmasını engelleyerek veri akışının geçmesini engeller ve sistem çokmüş olur. Şimdi dosyanın oluşturulması aşamasına geçelim. Dosyamı oluşturmak için herhangi bir metin belgesini masaüstünde açtıktan sonra bu dosyamızı programımızda çalıştırabilmek için uzantısını .tcl yapıyoruz. Sonrasında düğümleri oluşturuyoruz. Lms de yüklü olan 12.hafta notlarında bulunan simulator oluşturma, düğüm oluşturma ve topoloji oluşturma adımlarını YouTube den ve yapay zekadan destek alarak simülasyon için gerekli olan kodları yazdım.

Öncelikle bir simülatör nesnesi oluşturulur

Set ns [new Simulator] komutu ile

Simülasyon çıktısının kaydedileceği dosyaların oluşturulması

set nf [open out.nam w] $ns namtrace-all $nf

set f [open out.tr w]

$ns trace-all $tracefile

İlk satır 'out.nam' dosyasını yazmak üzere açar ve açılan dosyaya ‘nf’ etiketini verir. İkinci satırda, yukarıda oluşturduğumuz simülatör nesnesine, nam ile ilgili olacak olan bütün simülasyon verisini bu dosya içerisine yazmasını bildiririz.

Sonrasında düğümler oluşturulur;

set num\_nodes 25

for {set i 0} {$i < $num\_nodes} {incr i} {

set node($i) [$ns node]

}

Bir sonraki adım izleme dosyasını kapatan ‘finish’ prosedürünü ekleme yaparak simülasyon bitiş işlemlerini tanımlama ve nam programını başlatmaktır.

proc finish {} {

global ns nf f

$ns flush-traceclose

$nf exec nam out.nam &

exit 0

}

Bir sonraki satır, simülatör nesnesine simülasyon zamanından 5.0 sn sonra ‘finish’ prosedürünün çalıştırılmasını söyleyecektir.

$ns at 5.0 "finish"

Sadece kod satırına bakarak satırın işlevini anlamak mümkündür. Ns, ‘at’ komutuyla olayları oldukça basit bir şekilde programlamayı sağlar.

Nihayet son satır simülasyonu başlatır.

$ns run

Hocamızın 12.hafta slaytında bu kısma kadar yazılan kodlarla bu işlemin tamamlanıp simülasyonun çalışması beklenmektedir. Ama herhangi bir nesne (düğüm, link, vb.) veya olay tanımlanmadığı için 'nam: empty trace file out.nam' gibi bir hata mesajıyla karşılaşırız.

metin, ekran görüntüsü, yazılım, multimedya yazılımı içeren bir resim

Açıklama otomatik olarak oluşturuldu

Görüldüğü gibi 25 düğümlü ağ oluştu ve simülasyon test edildi.