

Ödevler: Linux ve Uygulamalar

Ertesi günün sabahına hazırlanması beklenen akşamlik ödevler

- Uptime komutunun çıktısında sonda bulunan sayılar ne anlama gelmektedir ?
oc@olmak2:~\$ uptime
23:56:28 up 1 day, 2:45, 4 users, load average: 0.06, 0.07, 0.07
oc@olmak2:~\$ _
- Aile soy ağacı tutan açık-kaynak bir veri tabanı uygulaması arıyorsunuz. Bu veri tabanı uygulaması MySQL gibi standart bir veritabanı uygulamasından farklı olmalı çünkü bu uygulama girilen bireylerin ve verilen ailevi bağılıkların (evli olmak, birinin oğlu veya dedesi olmak vb) mantıklı olup olmadığını da denetleyebilmelidir. Böyle bir uygulamayı google'da hangi kelimelerle aradınız ? Kullandığınız arama kelimelerinin sizi ulaştırdığı uygulama hangisi ? (En az bir uygulama bulunuz)
- Konu içinde “yedekle” ismiyle verilen kabuk betiğinin, yedekleme gerçekleştirirken yalnızca kütükleri değil aynı zamanda dizinleri de içermesini sağlayınız. Başka geliştirme fikirleriniz varsa savununuz ve ekleyiniz.
- Belirli bir işlevi yerine getiren aşağıdaki c/c++ kaynağının sonuç vermesi uzun zaman almaktadır. İşlev değişmeden kalmak şartıyla bu zamanı kısaltınız. (c/c++ dilinde “math.h” içinde, pi sayısı M_PI ve $a^b = \text{pow}(a, b)$ olarak karşılık bulur.)

```
#include <stdio.h>
#include <math.h>
int main() {
    float T, C1=300e-12, Ip=10e-9, N=120.0, R=1500.0, toplam = 0.0;
    int Ko, s;
    for (Ko=1 ; Ko<10000 ; Ko=Ko+1) {
        for (s=1 ; s<10000 ; s=s+1) {
            T = (pow((pow(((Ko*Ip)/(2*M_PI*C1*N)),0.5)),2)*((R*C1)*s+1))/
                (pow(s,2)/N+2*
                 (((R*C1)*(pow(((Ko*Ip)/(2*M_PI*C1*N)),0.5)))/2) *
                  s*(pow(((Ko*Ip)/(2*M_PI*C1*N)),0.5))/N +
                  pow((pow(((Ko*Ip)/(2*M_PI*C1*N)),0.5)),2)/N);
            toplam = toplam + T;
        }
    }
    printf("Toplam = %f \n", toplam);
    return 0;
}
```

Ödevler

Ödev Cevapları

- Uptime komutunun çıktısında sonda bulunan sayılar ne anlama gelmektedir ?
oc@olmak2:~\$ uptime
23:56:28 up 1 day, 2:45, 4 users, load average: 0.06, 0.07, 0.07
oc@olmak2:~\$ _
- Çıktının sonundaki üç sayı bilgisayarın tecrübe ettiği, son 1, 5 ve 15 dakikalık ortalama yük miktarlarıdır. “man uptime” komutu ile ilgili bilgiye ulaşılır.

```
UPTIME(1)                                Linux User's Manual                                UPTIME(1)

NAME
  uptime - Tell how long the system has been running.

SYNOPSIS
  uptime
  uptime [-V]

DESCRIPTION
  uptime gives a one line display of the following information. The current time, how long the system has been running, how many users are currently logged on, and the system load averages for the past 1, 5, and 15 minutes.

  This is the same information contained in the header line displayed by w(1).

  System load averages is the average number of processes that are either in a runnable or uninterruptable state. A process in a runnable state is either using the CPU or waiting to use the CPU. A process in uninterruptable state is waiting for some I/O access, eg waiting for disk. The averages are taken over the three time intervals. Load averages

Manual page uptime(1) line 1
```

Ödevler

Ödev Cevapları

- Konu içinde “yedekle” ismiyle verilen kabuk betiğinin, yedekleme gerçekleştirirken yalnızca kütükleri değil aynı zamanda dizinleri de içermesini sağlayınız. Başka geliştirme fikirleriniz varsa savununuz ve ekleyiniz.
- Betikte, kütükleri yaratılan dizinin içine çoğaltan (kopyalayan) komutun “cp” olduğu görülüyor. Bu komutun man sayfasına bakıldığında ve “dizinleri ve içerdiklerini de içer” anlamına gelen “recursive” kelimesini aradığımızda, ilgili anahtarın “-r” olduğu görülür. Dolayısıyla tek yapılması gereken cp komutuna “-r” anahtarını eklemektir.

```

--preserve[=ATTR_LIST]
    preserve the specified attributes (default: mode,owner-
    ship,timestamps), if possible additional attributes: context,
    links, all

--no-preserve=ATTR_LIST
    don't preserve the specified attributes

--parents
    use full source file name under DIRECTORY

-R, -r, --recursive
    copy directories recursively

--remove-destination
    remove each existing destination file before attempting to open
    it (contrast with --force)

--sparse=WHEN
    control creation of sparse files

--strip-trailing-slashes
    remove any trailing slashes from each SOURCE argument
Manual page cp(1) line 52/154 47%

```

```

File Edit Search Preferences Shell Macro Windows Help
1 #!/bin/bash
2
3 # Calıstırdığımız komutun ismine ulasiyoruz
4 echo " "
5 echo "Komutunuza verdiginiz isim : " $0
6
7 # Girilen anahtar sayisini denetliyoruz
8 if [ $# -eq 0 ] || [ $# -gt 1 ] ; then
9     echo " "
10    echo "HATA : Bir anahtar girilmeli."
11    echo " "
12    exit 2
13 fi
14
15 # Komutumuzun hangi anahtarları kabul
16 # ettiği sorulduğunda cevabimizi olusturuyoruz
17 if [ $1 == "-h" ] ; then
18    echo " "
19    echo "Icinde bulunulan dizindeki tum kutukleri"
20    echo "ve dizinleri, yeni <dizin ismi> dizinine"
21    echo "cogaltir ve *.tgz ile sikistirir. Kullanım sekli:"
22    echo $0 "<dizin ismi>"
23    echo " "
24    exit 1
25 fi
26
27 # Hersey yolunda, girilen anahtari
28 # kullaniciya gosteriyoruz
29 if [ $# -gt 0 ] ; then
30    echo " "
31    echo "Girdiginiz dizin ismi : " @$
32 fi
33
34 # Dizin var mi yok mu denetliyoruz; yoksa olusturuyoruz
35 # varsa karisiklikleri onlemek icin baska bir isim istiyoruz
36
37 if [ ! -e $1.tgz ] ; then
38    echo $1.tgz "yok, olusturuluyor..."
39    mkdir $1
40 else
41    echo "HATA : " $1.tgz "zaten var."
42    exit 2
43 fi
44
45 # Gorevi yerine getiriyoruz
46
47 for x in *
48 do
49     if [ $x != $1 ] ; then
50         cp -ravuf $x $1
51     fi
52 done
53
54 tar cfz $1.tgz $1
55 echo $1.tgz "bohcase olusturuldu."
56 rm -rf $1
57 echo "Gecici dizin" $1 "silindi."
58 echo "Komut tamamlandı."

```

Kütük: yedekle

Ödevler

Ödev Cevapları

- Aile soy ağacı tutan açık-kaynak bir veri tabanı uygulaması arıyorsunuz. Sözlüklerden, soy ağacı kavramına karşılık “*genealogy*” kelimesinin kullanıldığını öğreniyoruz. En çok kullanılan arama motorlarından olan “*google*” seçiliyor. Pek çok olasılıktan biri olarak, aranan kelime topluluğu, “*open source genealogy application*” olarak seçiliyor. Karşımıza çıkan bağlantılarda bir kaç dakikalık bir ilerleme ile, adı “*gramps*” olan bir uygulamaya ulaşıyoruz. Aynı aramayı “*free software project for genealogy*” olarak yaptığımızda ise hiç zaman kaybı olmaksızın karşımıza çıkan ilk bağlantı, gramps uygulamasının proje sayfası oluyor.

The screenshot shows a web browser window with the Google search engine. The search query is "free software project for genealogy". The search results are displayed below the search bar. The first result is "Project Planning Software" from www.planningforce.com. The second result is "Main Page - GRAMPS" from gramps-project.org/, which is highlighted by a red arrow. The third result is "Flexible Project Software" from www.instant-team.com. The fourth result is "Free Family Tree Software" from an unknown source. The browser's address bar shows the URL "http://www.google". The browser's menu bar includes File, Edit, View, History, Bookmarks, Tools, and Help. The browser's status bar at the bottom shows "Done".

Ödevler

Ödev Cevapları

- Belirli bir işlevi yerine getiren aşağıdaki programın sonuç vermesi uzun zaman almaktadır. İşlev değişmeden kalmak şartıyla bu zamanı kısaltınız.
- Kısa programa bakıldığında, 10000 kere 10000 defa hesaplanan ve T değişkenine atanan matematik işlev (fonksiyon) dışında iyileştirilebilecek başka bir şey görünmemektedir. Bu işlev, fazla sayıda CPU çevrimi gerektiren üst ve kare kök alma gibi işlemlerden oluşmaktadır ve denenebilecek tek şey, simgesel olarak T ifadesinin sadeleştirilmesi olabilir.
- Bir CAS uygulaması ile (*mathomatic* gibi) ifade sadeleştirilir, karşılık gelen C ifadesi yazılır ve kaynakta yerine konursa, yeni programın yaklaşık 26/3 kat hızlandırıldığı gözlemlenebilir.

```
oc@olmak2:~/Documents/HEP_Okulu/workDir$ date && ./kisalt && date
Fri Jan 2 06:05:53 CET 2009
Toplam = 24907114,000000
oc@olmak2:~/Documents/HEP_Okulu/workDir$ date && ./kisa && date
Fri Jan 2 06:06:19 CET 2009
Toplam = 24907114,000000
oc@olmak2:~/Documents/HEP_Okulu/workDir$ date && ./kisa && date
Fri Jan 2 06:06:26 CET 2009
Toplam = 24907114,000000
oc@olmak2:~/Documents/HEP_Okulu/workDir$
```

- "&&" özel işareti, kendisinden önce gelen komutun başarılı bir biçimde sonlanması durumunda kendisinden sonra gelen komutu işleme sokmak için kullanılmıştır. Kendisinden önce gelen komut başarısız olursa, kendisinden sonra gelen komut çalıştırılmaz; uygulama kesilir ve komut istemi satırına geri dönlür.

```
oc@olmak2:~/Documents/HEP_Okulu/workDir$ mathomatic controlTeorisi
Mathomatic version 14.0.6 (www.mathomatic.org)
Copyright (C) 1987-2008 George Gesslein II.
This is free software with NO WARRANTY.

100 equation spaces available, 960 kilobytes per equation space.
ANSI color mode enabled.
1-> T=(((Ko*Ip)/(2*pi*C1*N)^.5)^2 * ((R*C1)*s+1))/(s^2/N+2*(((R*C1)*((Ko*Ip)/(2*
pi*C1*N)^.5))/2)*s*(((Ko*Ip)/(2*pi*C1*N)^.5)/N+(((Ko*Ip)/(2*pi*C1*N)^.5)^2/N)

#1: T =

-----
Ko*Ip*((R*C1*s) + 1)
-----
R*C1*((s^2) / (2*pi*C1*N)^2) * s * (Ko*Ip / (2*pi*C1*N)^2) + Ko*Ip / (2*pi*C1*N)

1-> simplify

#1: T =
-----
Ko*Ip*N*((R*C1*s) + 1)
-----
((2*C1*(s^2)*pi*N) + (Ko*Ip*((C1*s*R) + 1)))

1-> code
T = (Ko*Ip*N*((R*C1*s) + 1.0)/((2.0*C1*(s*s)*M_PI*N) + (Ko*Ip*((C1*s*R) + 1.0)))));
1-> quit
oc@olmak2:~/Documents/HEP_Okulu/workDir$
```

```
New Open Save Print... Undo Redo Cut Copy Paste Find Replace
*kisa.c
1 #include<stdio.h>
2 #include<math.h>
3
4 int main() {
5
6     float T, C1=300e-12, Ip=10e-9, N=120.0, R=1500.0, toplam =
0.0;
7     int Ko, s;
8     for (Ko=1; Ko<10000; Ko=Ko+1) {
9         for (s=1; s<10000; s=s+1) {
10            T = (Ko*Ip*N*((R*C1*s)+1.0)/
11                ((2.0*C1*(s*s)*M_PI*N) +
12                 (Ko*Ip*((C1*s*R) + 1.0)))));
13            toplam = toplam + T;
14        }
15    }
16
17    printf("Toplam = %f \n", toplam);
18
19    return 0;
20}
```