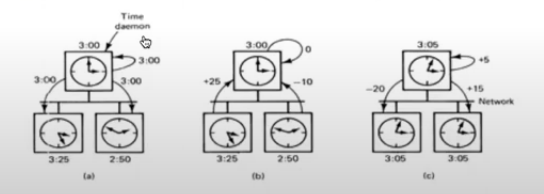
<https://www.google.com/search?q=berkeley+algorithm&source=lnms&tbm=vid&sa=X&ved=2ahUKEwibyo7g187-AhVXcmwGHcvrC6cQ_AUoAnoECAEQBA&biw=1517&bih=694&dpr=0.9#fpstate=ive&vld=cid:1cd4e59d,vid:rYWxIB8HoRw>



//Practical : 4 . Implement Berkeley

//Aim – . Implement Berkeley algorithm for clock synchronization.

//Program to demonstrate Berkeley clock synchronization algorithm

import java.io.\*;

import java.util.\*;

public class Berkley

{

float diff(int h, int m, int s, int nh, int nm, int ns){

int dh = h-nh;

int dm = m-nm;

int ds = s-ns;

int diff = (dh\*60\*60)+(dm\*60)+ds;

return diff;

}

float average(float diff[], int n){

int sum=0;

for(int i=0; i<n; i++)

{

sum+=diff[i];

}

float average = (float)sum/(n+1);

System.out.println("The average of all time differences is "+average);

return average;

}

void sync(float diff[], int n, int h, int m, int s, int nh[], int nm[], int ns[], float average)

{

for(int i=0;i<n;i++)

{

diff[i]+=average;

int dh=(int)diff[i]/(60\*60);

diff[i]%=(60\*60);

int dm=(int)diff[i]/60;

diff[i]%=60;

int ds=(int)diff[i];

nh[i]+=dh; //assign different hours value in “nh” variable

if(nh[i]>23)

{

nh[i]%=24;

}

nm[i]+=dm; //assign different minutes value in “nm” variable

if(nm[i]>59)

{

nh[i]++;

nm[i]%=60;

}

ns[i]+=ds; //assign different second value in “ns” variable

if(ns[i]>59)

{

nm[i]++;

ns[i]%=60;

}

if(ns[i]<0)

{

nm[i]--;

ns[i]+=60;

}

}

h+=(int)(average/(60\*60));

if(h>23)

{ h%=24;

}

m+=(int)(average/(60\*60\*60));

if(m>59)

{

h++;

m%=60;

}

s+=(int)(average%(60\*60\*60));

if(s>59)

{

m++;s%=60;

}

if(s<0)

{

m--;

s+=60;

}

System.out.println("The synchronized clocks are:\nTime Server ---> "+h+" : "+m+" : "+s);

for(int i=0;i<n;i++)

{

System.out.println("Node "+(i+1)+" ---> "+nh[i]+" : "+nm[i]+" : "+ns[i]);

}

}

public static void main(String[] args) throws IOException {

Berkley b = new Berkley();

Date date = new Date();

BufferedReader obj = new BufferedReader(new InputStreamReader(System.in));

System.out.println("Enter number of nodes:");

int n = Integer.parseInt(obj.readLine()); int h = date.getHours();

int m = date.getMinutes();

int s = date.getSeconds();

int nh[] = new int[n];

int nm[] = new int[n];

int ns[] = new int[n];

for(int i=0; i<n; i++)

{

System.out.println("Enter time for node "+(i+1)+"\n Hours:");

nh[i]=Integer.parseInt(obj.readLine());

System.out.println("Minutes:");

nm[i]=Integer.parseInt(obj.readLine());

System.out.println("Seconds:");

ns[i]=Integer.parseInt(obj.readLine());

}

for(int i=0; i<n; i++)

{

System.out.println("Time Server sent time "+h+" : "+m+" : "+s+" to node "+(i+1));

}float diff[] = new float[n];

for(int i=0;i<n;i++)

{

diff[i] = b.diff(h,m,s,nh[i],nm[i],ns[i]);

System.out.println("Node "+(i+1)+" sent time difference of "+(int)diff[i]+" to Time Server.");

}

float average = b.average(diff,n);

b.sync(diff, n, h, m, s, nh, nm, ns, average);

}

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

[c:\javac](file:///C:\c:\javac) Berkley.java

[c:\java](file:///C:\c:\java) Berkle

