МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

НАВЧАЛЬНО-НАУКОВИЙ КОМПЛЕКС

«ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ»

Лабораторна робота № 2

з курсу «Системний аналіз»

Тема: «Принципи рішень неструктурованих проблем. Методи парних і послідовних порівнянь»

Виконав:

студент IV курсу

групи ДА-32

Колінько Анжела

Київ – 2017

Варіант 11

Мета:

Освоїти принципи вирішення неструкртурованих проблем. Вивчити методи:

парних порівнянь і послідовних порівнянь.

зважування експертних оцінок

переваги

рангу

повного попарного порівняння

Завдання: 11%7=4

Варіант 4

*Методи парних і послідовних порівнянь*

З республіканського і місцевих бюджетів виділено кошти у сферу охорони здоров'я експерт проводить оцінку найбільш потребуючої і важливої сфери медицини для отримання субсидій.

1. Заміна всього обладнання, що відпрацювало нормативний термін, на нове;

2. Встановлення дорогого сучасного обладнання в спеціалізованих центрах і диспансерах;

3. Відкриття поліклінік у густонаселених мікрорайонах;

4. Будівництво наркологічного центру;

Оцінки експерта запропонованих цілей наведені в матриці:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Zj | Z2 | Z3 | Z4 | Z1 |
| pi | 100 | 68 | 40 | 33 |

Де Z1...j — цілі.

Визначити найбільш важливу ціль.

*Метод зважування експертних оцінок*

Два експерта проводять оцінку 4-х цілей, які пов'язані з вирішенням транспортної проблеми в густо заселеному новому районі столиці.

1. Побудувати метрополітен

2. Придбати 2-поверховий автобус

3. Розширити транспортну мережу

4. Ввести швидкісний трамвай

В результаті проведення експертизи отримана матриця ваг цілей:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Еj/Zi | Z1 | Z2 | Z3 | Z4 |
| Е1 | 0,2 | 0,14 | 0,16 | 0,5 |
| Е2 | 0,4 | 0,25 | 0,15 | 0,2 |

Е1 — головний архітектор столиці.

Е2 — голова комітету із влаштування по місту.

Оцінки компетентності, базуються на стажі роботи, рівні R1 = 8 і R2 = 8,5.

*Метод переваги*

Для вирішення проблеми, пов'язаної з неможливістю надання житла іногороднім студентам була скликана група експертів з 4-х чоловік.

Запропоновано кілька альтернатив:

1. Побудувати новий гуртожиток

2. Зняти багатоквартирний будинок і частково оплачувати житло

3. Призначити доплату незаселеним студентам

Оцінки експертів запропонованих альтернатив наведені в матриці

|  |  |  |  |
| --- | --- | --- | --- |
| Еj/Zi | Z1 | Z2 | Z3 |
| Е1 | 1 | 2 | 3 |
| Е2 | 2 | 1 | 3 |
| Е3 | 2 | 3 | 1 |
| Е4 | 3 | 2 | 1 |
| Е5 | 3 | 1 | 2 |
| Е6 | 1 | 2 | 3 |

Де Е1...i — експерти, Z1...j — проекти

Визначити найкращу альтернативу.

*Mетод рангу*

Для вирішення проблеми, пов'язаної з неможливістю надання житла іногороднім студентам була скликана група експертів з 4-х чоловік.

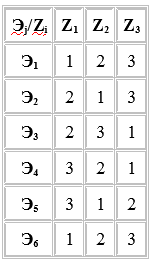
Запропоновано кілька альтернатив:

1. Побудувати новий гуртожиток

2. Зняти багатоквартирний будинок і частково оплачувати житло

3. Призначити доплату незаселеним студентам

Оцінки експертів запропонованих альтернатив наведені в матриці

Визначити найкращу альтернативу

*Метод повного попарного порівняння*

Телеканал ОНТ прийняв проект нового ТВ шоу «в реальному часі», у зв’язку з чим постало питання виділення ефірного часу. Двом телеекспертам були запропоновані такі альтернативи:

Z1 — показувати найбільш цікаві моменти 20 хв. 3 рази на добу з понеділка по п’ятницю

Z2 — формувати звіти за неділю та пускати у вечірньому випуску по вихідним

Z3 — показувати кожен день «щоденник подій» та велику передачу в суботу ввечері

Матриці попарного порівняння альтернатив(розмір шкали 16):

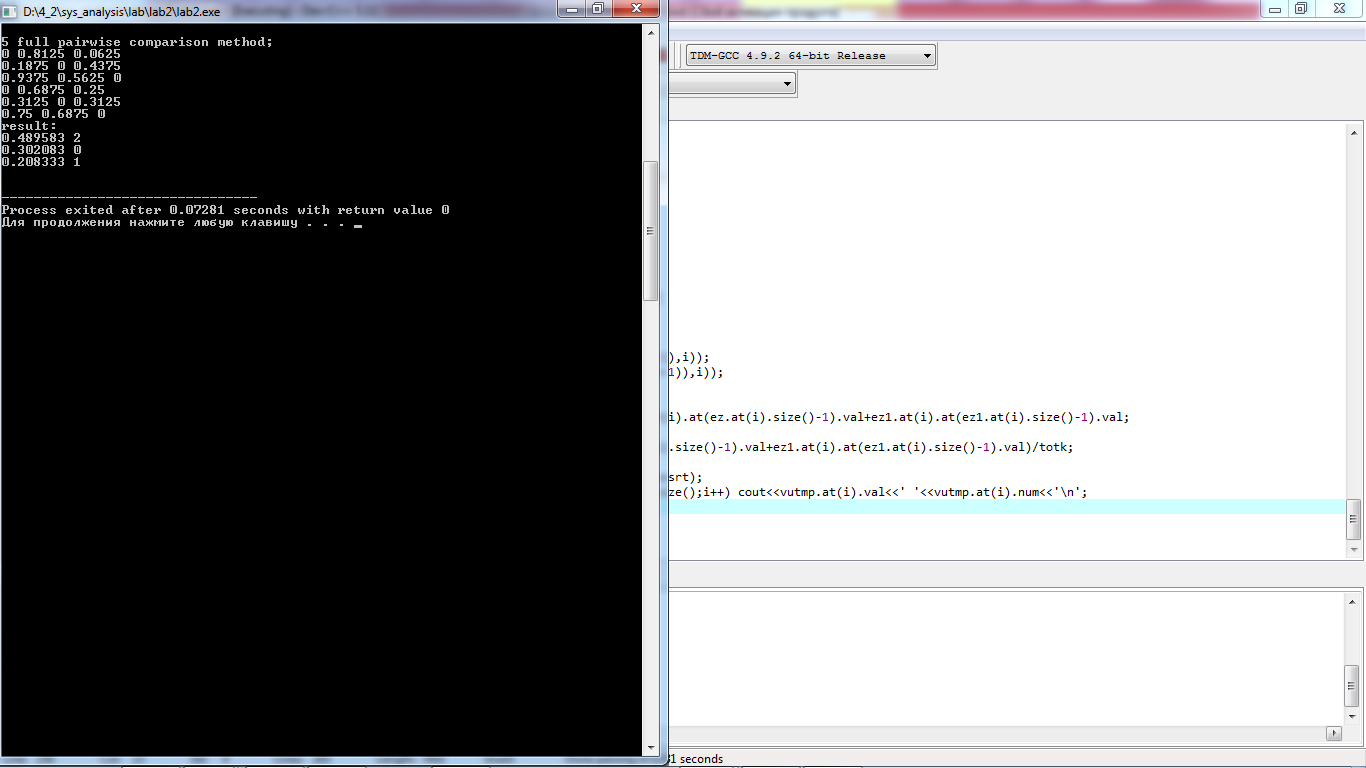
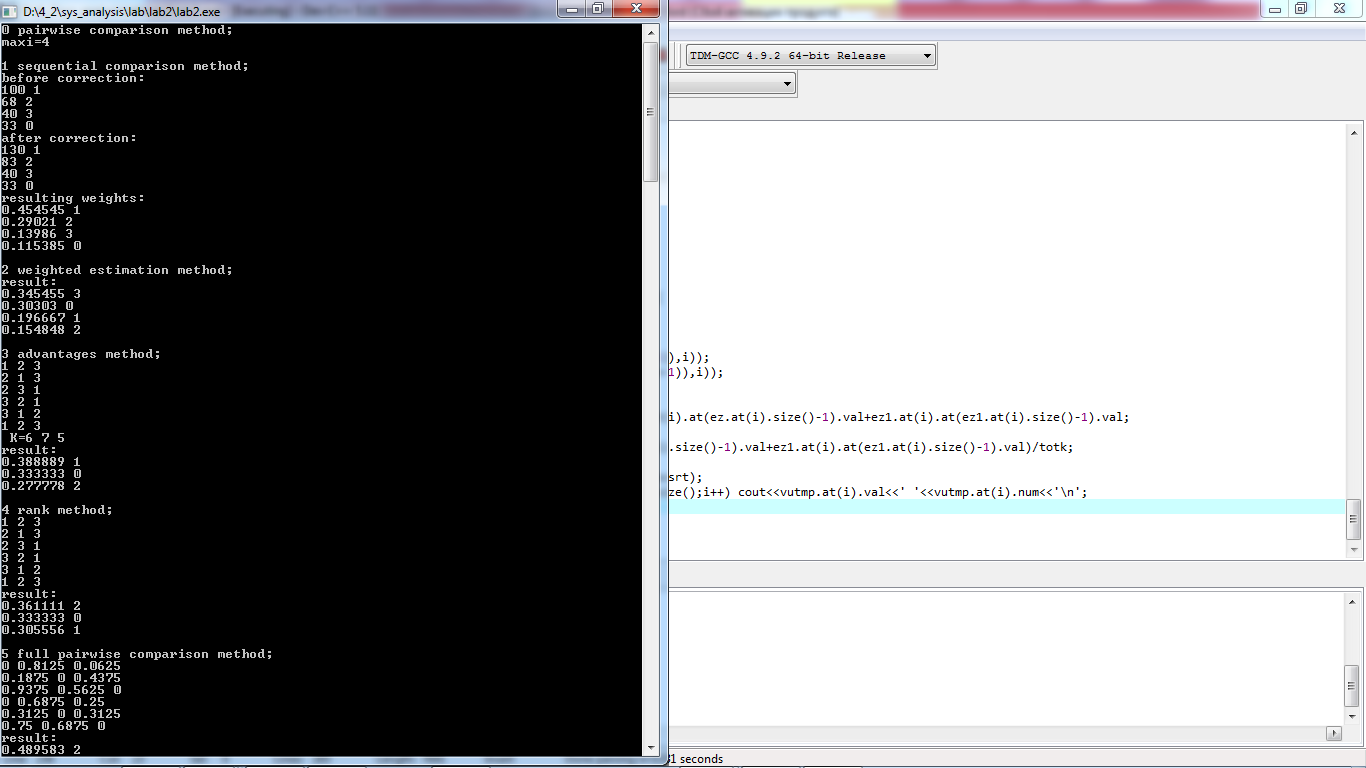
|  |  |  |  |
| --- | --- | --- | --- |
| Е1 | Z1 | Z2 | Z3 |
| Z1 |  | 13/16 | 1/16 |
| Z2 | 3/16 |  | 7/16 |
| Z3 | 15/16 | 9/16 |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Е2 | Z1 | Z2 | Z3 |
| Z1 |  | 11/16 | 4/16 |
| Z2 | 5/16 |  | 5/16 |
| Z3 | 12/16 | 11/16 |  |

де Е1...i — експерти, Z1...j — об’єкти експертизи

Вибрати найкращу альтернативу

Результати роботи програми:



Висновок: на лабораторній роботі розглядались методи парних, послідовних порівнянь, зважування експертних оцінок, переваги, рангу, повного попарного порівняння.

Додаток. Лістинг.

#include<iostream>

#include<vector>

#include<algorithm>

#define dltst 30

using namespace std;

typedef unsigned short int usi;

struct usib{

double val;

usi num;

usib(){}

usib(double nval, usi nnum): val(nval), num(nnum){}

};

bool dscsrt(usib i,usib j){

return(i.val>j.val);

}

void getubv(double darr[],vector<usib>& ubarr,usi n){

ubarr.clear();

for(usi i=0;i<n;i++){

usib ubtmp; ubtmp.val=darr[i];

ubarr.push\_back(ubtmp);

}

}

int main(){

usi cnt=0;

cout<<cnt<<" pairwise comparison method;\n";

vector<usi> z; usi z1[]={33,100,68,40}, maxi=0;

z.assign(z1,z1+(sizeof(z1)/sizeof(z1[0])));

for(usi i=0;i<z.size();i++){

for(usi j=1;j<z.size();j++){

if(i>j){

if(z.at(i)<z.at(j)){

maxi=i;

}

}

}

}

cout<<"maxi="<<maxi+1<<"\n\n"; cnt++;

//system("pause");

cout<<cnt<<" sequential comparison method;\n";

vector<usib> zz;

for(usi i=0;i<z.size();i++){

usib usibtmp; usibtmp.val=z.at(i); usibtmp.num=i;

zz.push\_back(usibtmp);

}

stable\_sort(zz.begin(),zz.end(),dscsrt);

cout<<"before correction:\n";

for(usi i=0;i<zz.size(); i++){

cout<<zz.at(i).val<<' '<<zz.at(i).num<<'\n';

}

usi dlt=dltst;

for(usi i=0;i<zz.size();i++){

for(usi j=1;j<zz.size();j++){

for(usi k=2;k<zz.size();k++){

if(i<j&&j<k&&zz.at(i).val<(zz.at(j).val+zz.at(k).val)){

zz.at(i).val+=dlt; dlt/=2;

}

}

}

}

unsigned int tots=0;

cout<<"after correction:\n";

for(usi i=0;i<zz.size();i++){

cout<<zz.at(i).val<<' '<<zz.at(i).num<<'\n';

tots+=zz.at(i).val;

}

for(usi i=0;i<zz.size();i++){

zz.at(i).val/=tots;

}

cout<<"resulting weights:\n"; for(usi i=0;i<zz.size();i++) cout<<zz.at(i).val<<' '<<zz.at(i).num<<'\n';

cout<<'\n'; cnt++;

cout<<cnt<<" weighted estimation method;\n";

vector<usib> iz1,iz2; double r1=8,r2=8.5,totr,tmps=0; bool isfst=1;

totr=r1+r2; r1/=totr; r2/=totr;

double dz1[]={0.2,0.14,0.16,0.5},dz2[]={0.4,0.25,0.15,0.2};

for(usi i=0;i<sizeof(dz1)/sizeof(dz1[0]);i++){

usib usibtmp; usibtmp.val=dz1[i]; usibtmp.num=i;

iz1.push\_back(usibtmp);

usibtmp.val=dz2[i];

iz2.push\_back(usibtmp);

}

/\*for(usi i=0;i<iz1.size();i++){

cout<<iz1.at(i).val<<' '<<iz2.at(i).val<<'\n';

}\*/

for(usi i=0;i<iz1.size();i++){

if(isfst){

tmps+=iz1.at(i).val;

if(i==iz1.size()-1){

if(tmps!=1) {cout<<"\nerr: tmps1!=1\n"; system("pause");}

isfst=0; i=-1; tmps=0;

}

}

else{

tmps+=iz2.at(i).val;

if(i==iz2.size()-1){

if(tmps!=1) {cout<<"\nerr: tmps2!=1\n"; system("pause");}

isfst=1; tmps=0;

}

}

}

for(usi i=0;i<iz1.size();i++){

iz1.at(i).val=iz1.at(i).val\*r1+iz2.at(i).val\*r2;

}

stable\_sort(iz1.begin(),iz1.end(),dscsrt);

cout<<"result:\n"; for(usi i=0;i<iz1.size();i++) cout<<iz1.at(i).val<<' '<<iz1.at(i).num<<'\n';

cout<<'\n'; cnt++;

cout<<cnt<<" advantages method;\n";

vector<vector<usib>> ez; vector<usib> vutmp; double darr[3],totk=0;

darr[0]=1;darr[1]=2;darr[2]=3;

getubv(darr,vutmp,sizeof(darr)/sizeof(darr[0]));

ez.push\_back(vutmp);

darr[0]=2;darr[1]=1;darr[2]=3;

getubv(darr,vutmp,sizeof(darr)/sizeof(darr[0]));

ez.push\_back(vutmp);

darr[0]=2;darr[1]=3;darr[2]=1;

getubv(darr,vutmp,sizeof(darr)/sizeof(darr[0]));

ez.push\_back(vutmp);

darr[0]=3;darr[1]=2;darr[2]=1;

getubv(darr,vutmp,sizeof(darr)/sizeof(darr[0]));

ez.push\_back(vutmp);

darr[0]=3;darr[1]=1;darr[2]=2;

getubv(darr,vutmp,sizeof(darr)/sizeof(darr[0]));

ez.push\_back(vutmp);

darr[0]=1;darr[1]=2;darr[2]=3;

getubv(darr,vutmp,sizeof(darr)/sizeof(darr[0]));

ez.push\_back(vutmp);

for(usi i=0;i<ez.size();i++){

for(usi j=0; j<ez.at(i).size();j++){

cout<<ez.at(i).at(j).val<<' ';

ez.at(i).at(j).val=ez.at(i).size()-ez.at(i).at(j).val;

ez.at(i).at(j).num=j;

vutmp.at(j).val=0; vutmp.at(j).num=j;

//cout<<" q "<<ez.at(i).at(j).val<<' ';

}

cout<<'\n';

}

for(usi i=0;i<ez.size();i++){

for(usi j=0; j<ez.at(i).size();j++){

vutmp.at(j).val+=ez.at(i).at(j).val;

}

}

cout<<" K=";

for(usi j=0; j<vutmp.size();j++){

cout<<vutmp.at(j).val<<' ';

totk+=vutmp.at(j).val;

}

for(usi i=0;i<vutmp.size();i++){

vutmp.at(i).val/=totk;

}

stable\_sort(vutmp.begin(),vutmp.end(),dscsrt);

cout<<"\nresult:\n"; for(usi i=0;i<vutmp.size();i++) cout<<vutmp.at(i).val<<' '<<vutmp.at(i).num<<'\n';

cout<<'\n'; cnt++;

cout<<cnt<<" rank method;\n";

ez.clear(); vutmp.clear(); totk=0;

darr[0]=1;darr[1]=2;darr[2]=3;

getubv(darr,vutmp,sizeof(darr)/sizeof(darr[0]));

ez.push\_back(vutmp);

darr[0]=2;darr[1]=1;darr[2]=3;

getubv(darr,vutmp,sizeof(darr)/sizeof(darr[0]));

ez.push\_back(vutmp);

darr[0]=2;darr[1]=3;darr[2]=1;

getubv(darr,vutmp,sizeof(darr)/sizeof(darr[0]));

ez.push\_back(vutmp);

darr[0]=3;darr[1]=2;darr[2]=1;

getubv(darr,vutmp,sizeof(darr)/sizeof(darr[0]));

ez.push\_back(vutmp);

darr[0]=3;darr[1]=1;darr[2]=2;

getubv(darr,vutmp,sizeof(darr)/sizeof(darr[0]));

ez.push\_back(vutmp);

darr[0]=1;darr[1]=2;darr[2]=3;

getubv(darr,vutmp,sizeof(darr)/sizeof(darr[0]));

ez.push\_back(vutmp);

for(usi i=0;i<ez.size();i++){

totk=0;

for(usi j=0; j<ez.at(i).size();j++){

cout<<ez.at(i).at(j).val<<' ';

totk+=ez.at(i).at(j).val;

ez.at(i).at(j).num=j;

vutmp.at(j).val=0; vutmp.at(j).num=j;

}

for(usi j=0; j<ez.at(i).size();j++){

ez.at(i).at(j).val/=totk;

}

cout<<'\n';

}

for(usi j=0; j<ez.at(0).size();j++){

totk=0;

for(usi i=0;i<ez.size();i++){

totk+=ez.at(i).at(j).val;

}

vutmp.at(j).val=totk/ez.size();

}

stable\_sort(vutmp.begin(),vutmp.end(),dscsrt);

cout<<"result:\n"; for(usi i=0;i<vutmp.size();i++) cout<<vutmp.at(i).val<<' '<<vutmp.at(i).num<<'\n';

cout<<'\n'; cnt++;

cout<<cnt<<" full pairwise comparison method;\n";

ez.clear(); vutmp.clear(); totk=0; vector<vector<usib>> ez1; double totk1=0; usi n=3;

darr[0]=0;darr[1]=13.0/16;darr[2]=1.0/16;

getubv(darr,vutmp,sizeof(darr)/sizeof(darr[0]));

ez.push\_back(vutmp);

darr[0]=3.0/16;darr[1]=0;darr[2]=7.0/16;

getubv(darr,vutmp,sizeof(darr)/sizeof(darr[0]));

ez.push\_back(vutmp);

darr[0]=15.0/16;darr[1]=9.0/16;darr[2]=0;

getubv(darr,vutmp,sizeof(darr)/sizeof(darr[0]));

ez.push\_back(vutmp);

darr[0]=0;darr[1]=11.0/16;darr[2]=4.0/16;

getubv(darr,vutmp,sizeof(darr)/sizeof(darr[0]));

ez1.push\_back(vutmp);

darr[0]=5.0/16;darr[1]=0;darr[2]=5.0/16;

getubv(darr,vutmp,sizeof(darr)/sizeof(darr[0]));

ez1.push\_back(vutmp);

darr[0]=12.0/16;darr[1]=11.0/16;darr[2]=0;

getubv(darr,vutmp,sizeof(darr)/sizeof(darr[0]));

ez1.push\_back(vutmp);

for(usi i=0;i<ez.size();i++){

for(usi j=0; j<ez.at(i).size();j++){

cout<<ez.at(i).at(j).val<<' ';

ez.at(i).at(j).num=j;

vutmp.at(j).val=0; vutmp.at(j).num=j;

}

cout<<'\n';

}

for(usi i=0;i<ez1.size();i++){

for(usi j=0; j<ez1.at(i).size();j++){

cout<<ez1.at(i).at(j).val<<' ';

ez1.at(i).at(j).num=j;

}

cout<<'\n';

}

for(usi i=0;i<ez.size();i++){

totk=0; totk1=0;

for(usi j=0; j<ez.at(i).size();j++){

totk+=ez.at(i).at(j).val;

totk1+=ez1.at(i).at(j).val;

}

ez.at(i).push\_back(usib(totk/(n\*(n-1)),i));

ez1.at(i).push\_back(usib(totk1/(n\*(n-1)),i));

}

totk=0;

for(usi i=0;i<ez.size();i++) totk+=ez.at(i).at(ez.at(i).size()-1).val+ez1.at(i).at(ez1.at(i).size()-1).val;

for(usi i=0;i<vutmp.size();i++){

vutmp.at(i).val=(ez.at(i).at(ez.at(i).size()-1).val+ez1.at(i).at(ez1.at(i).size()-1).val)/totk;

}

stable\_sort(vutmp.begin(),vutmp.end(),dscsrt);

cout<<"result:\n"; for(usi i=0;i<vutmp.size();i++) cout<<vutmp.at(i).val<<' '<<vutmp.at(i).num<<'\n';

return 0;

}