Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

ОТЧЕТ

к лабораторной работе по дисциплине:

|  |
| --- |
| **Методы и средства проектирования информационных систем и технологий** |
| Поддержка принятия решений в задачах группового (коллективного) выбора |

наименование темы

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Выполнил студент группы: | ИСМб-19-1 |  |  |  | Малиновцев И.А. |
|  |  | подпись |  | Фамилия И.О |
| Проверил: |  |  |  |  | Юрин А.Ю. |
|  | должность |  | подпись |  | Фамилия И.О |

Иркутск 2022 г.

Содержание

[1 Постановка задачи 3](#_Toc118802954)

[2 Описание тестовых примеров 4](#_Toc118802955)

[3 Описание реализуемого метода группового выбора 6](#_Toc118802956)

[3.1 Неформальное описание 6](#_Toc118802957)

[3.2 Блок-схема 6](#_Toc118802958)

[4 Результаты решения тестовых задач 7](#_Toc118802959)

[5 Фрагмент сгенерированной документации 8](#_Toc118802960)

[6 Листинг кода 9](#_Toc118802961)

1 Постановка задачи

Цель работы: выработка и закрепление навыков по использованию систем и модулей поддержки принятия решений в задачах группового (коллективного) выбора.

Для выбранного варианта задания необходимо:

1. Разработать 6 тестовых примеров для задач группового (коллективного) выбора, при этом 2 примера должны содержать парадоксы голосования.

2. Решить разработанные примеры с использованием СППР КВ.

3. Произвести разработку собственного программного модуля, реализующего метод (согласно варианту задания).

4. Решить примеры с использованием разработанного модуля в составе СППР КВ.

5. Сгенерировать комплект документации (отчет по решенным задачам).

**Вариант**

|  |  |  |
| --- | --- | --- |
| N | Метод | Краткое описание |
| 6. | процедура Симпсона | для каждой альтернативы определяется функция, равная минимальному числу участников, предпочитающих данную альтернативу любой другой, отношение группового предпочтения строится по убыванию значений функции |

2 Описание тестовых примеров

Для выполнения данной рабораторной работы было составлено шесть тестовых примеров. Два примера, согласно заданию, содержали парадоксы.

**Пример 1**

Выбор автомобиля.

**Описание:**

Выбор автомобиля для комфортного передвижения.

**Альтернативы:**

a=Tayota Prius

b= Chevrolet Niva

c=Lada Granta

**Ранжировки:**

a-b-c=12

b-a-c=10

с-a-b=2

**Пример 2**

Выбор жанра книги.

**Описание:**

Выбор жанра книги для прочтения.

**Альтернативы:**

a=Ужас

b=Фантастика

c=Фентези

**Ранжировки:**

c-a-b=12

a-c-b=7

c-b-a=3

**Пример3**

Выбор телефона.

**Описание:**

Выбор телефона с учетом ценновой политики.

**Альтернативы:**

a=Infinix Note 12 2023

b= Samsung Galaxy A13

c= Honor X7

**Ранжировки:**

c-b-a=16

c-a-b=12

b-c-a=10

**Пример 4**

Выбор мышки

**Описание:**

Задача подбора мышки с учетом наличия дополнительных кнопок и эргономики.

**Альтернативы:**

a=Razer DeathAdder Essential

b=SteelSeries Rival 3

c=Logitech G102 LIGHTSYNC

**Ранжировки:**

a-b-c=16

b-a-c=14

c-a-b=13

**Пример 5 (Парадокс)**

Выбор вуза

**Описание:**

Выбор учебного завидения для поступления.

**Альтернативы:**

a=ИГУ

b=ИРНИТУ

c=ИРНИТУ

**Ранжировки:**

a-с-b=20

b-c-a=15

c-b-a=17

**Пример 6 (Парадокс)**

Выбор производителя процессора

**Описание:**

Выбор производителя процессора.

**Альтернативы:**

a=Intel

b=AMD

c=Intel

**Ранжировки:**

a-с-b=21

b-c-a=17

c-b-a=12

3 Описание реализуемого метода группового выбора

3.1 Неформальное описание

Цель данной процедуры – стремление сделать манимальный результат максимальным, т. е. получить наилучшую выгоду в наихудших условиях. В данном случае максиминный критерий — критерий, согласно которому происходит стремление получения максимального выигрыша в наихудшей ситуации.

3.2 Блок-схема

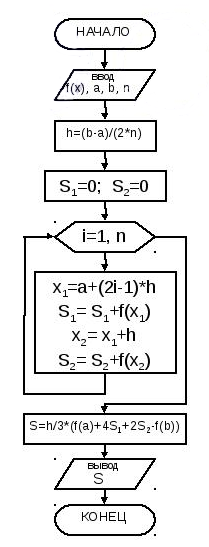


Рисунок 1 – Блок-схема метода

4 Результаты решения тестовых задач

В ходе решения тестовых примеров методом Симпсона с помощью программного средства СППР КВ и соответствующей готовой библиотеки были получены следующие результаты.

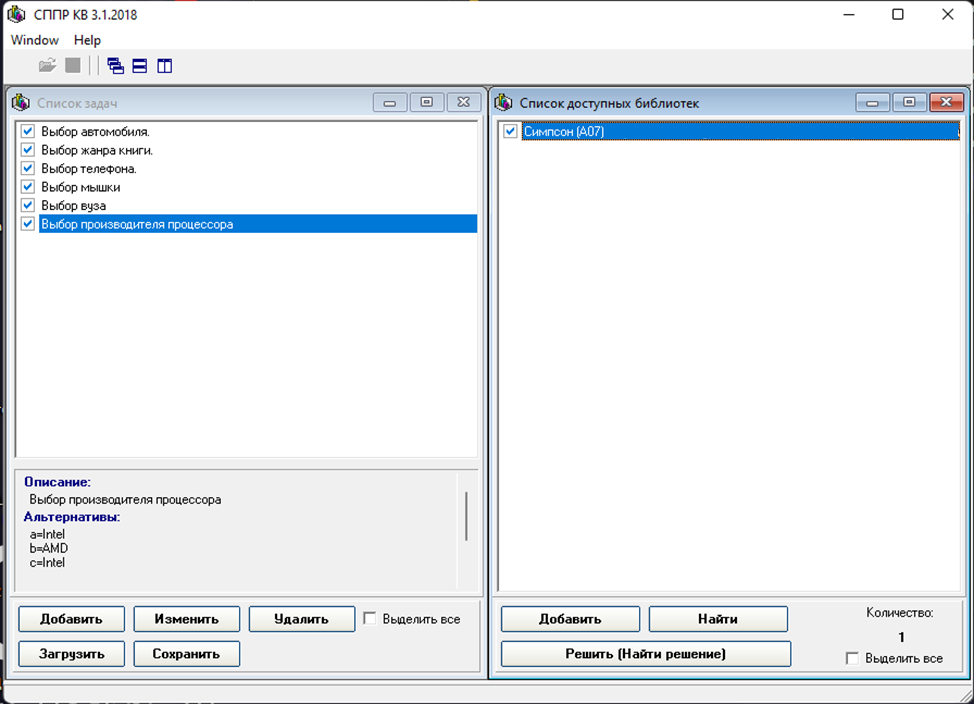


Рисунок 2 – Загрузка тестовых решений и метода Симпсона

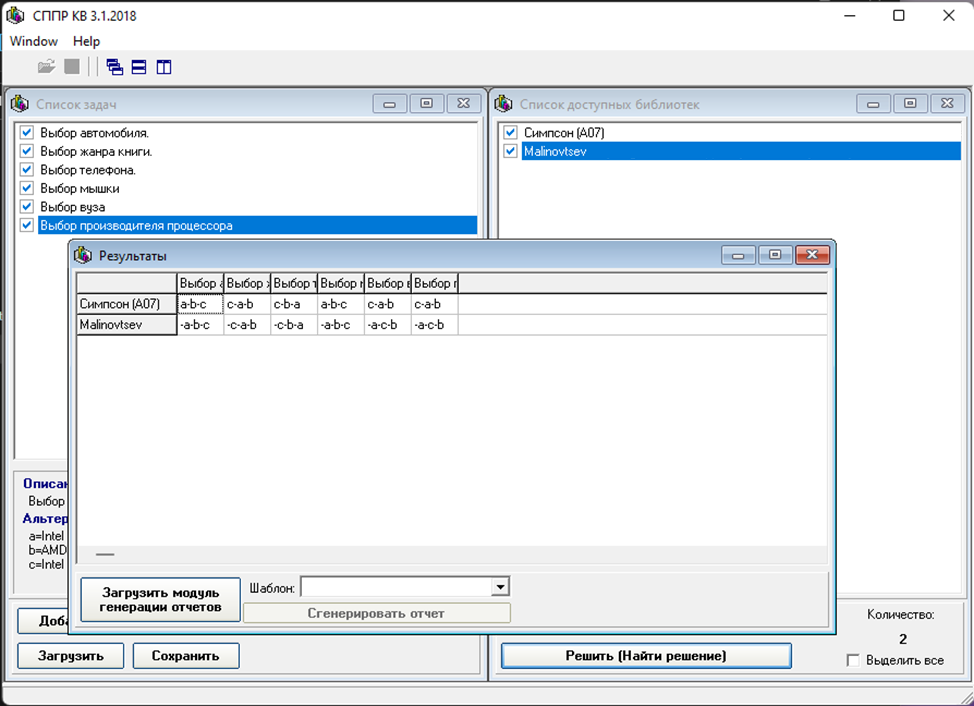


Рисунок 3 – Результат работы библиотек

В результате решения было выявлено, что разработанная библиотека выдает аналогичный результат в четырех из шести представленных задачах. Погрешности были выявлены в задачах-парадоксах.

5 Фрагмент сгенерированной документации

**Отчет № 08.11.2022**

**по задачам и методам**

Таблица 1 – Отчет о работе системы

|  |  |  |
| --- | --- | --- |
| Задача/ Метод | Симпсон (A07) | Malinovtsev |
| Выбор автомобиля. | a-b-c | -a-b-c |
| Выбор жанра книги. | c-a-b | -c-a-b |
| Выбор телефона. | c-b-a | -c-b-a |
| Выбор мышки | a-b-c | -a-b-c |
| Выбор вуза | c-a-b | -a-c-b |
| Выбор производителя процессора | c-a-b | -a-c-b |

Количество задач: 6

Количество методов: 2

6 Листинг кода

library MyDll1;

{ Important note about DLL memory management: ShareMem must be the

first unit in your library's USES clause AND your project's (select

Project-View Source) USES clause if your DLL exports any procedures or

functions that pass strings as parameters or function results. This

applies to all strings passed to and from your DLL--even those that

are nested in records and classes. ShareMem is the interface unit to

the BORLNDMM.DLL shared memory manager, which must be deployed along

with your DLL. To avoid using BORLNDMM.DLL, pass string information

using PChar or ShortString parameters. }

uses

SysUtils,

Classes;

{$R \*.res}

function Execute (const data: WideString): WideString ; stdcall ;

var

i,j : Integer;

tmTs,tmTs1 : TStringList;

tmA : TStringList;

s,s1,s2 : ShortString;

begin

//

tmTs:=TStringList.Create;

tmTs1:=TStringList.Create;

tmA:=TStringList.Create;

tmTs.Text:=data;

for i:=0 to tmTs.Count-1 do

begin

tmTs1.Delimiter:='-';

tmTs1.DelimitedText:=tmTs.Names[i];

s2:=tmTs.ValueFromIndex[i];

for j:=0 to tmTs1.Count-1 do

begin

s:=tmTs1.Strings[j];

if tmA.IndexOfName(s)=-1 then

tmA.Add(s+'=0');

if j=tmTs1.Count-1 then

begin

s1:=tmA.Values[s];

tmA.Values[s]:=IntToStr(StrToInt(s1)+StrToInt(s2));

end;

end;

end;

For i:=0 to tmA.Count-2 do

For j:=0 to tmA.Count-2 do

if tmA.ValueFromIndex[i]<= tmA.ValueFromIndex[j+1] then

tmA.Exchange(j+1,j+1)

else

tmA.Exchange(j, j);

Result:='';

For i:=0 to tmA.Count-1 do

Result:=Result+'-'+tmA.Names[i];

end;

function About: WideString; stdcall;

begin

Result := PChar('Malinovtsev');

end;

exports

About,

Execute;

begin

end.