Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

Специальность 09.02.03 «Программирование в компьютерных системах»

**ОТЧЕТ**

**ПО УЧЕБНОЙ ПРАКТИКЕ**

Профессиональный модуль ПМ.01 Разработка модулей программного обеспечения для компьютерных систем

*(наименование профессионального модуля)*

Выполнил:

обучающийся учебной группы № 120

Афонин Г.Д.

*(И.О. Фамилия)*

Проверил:

руководитель практики от колледжа:

Т.Г. Аксёнова

*(И.О. Фамилия)*

М.В. Морозова

*(И.О. Фамилия)*

**Москва**

**20­­­­22**

**Перечень заданий/работ, выполненных в ходе учебной практики**

Специальность 09.02.03 «Программирование в компьютерных системах»

*(наименование специальности)*

Профессиональный модуль ПМ.01 Разработка модулей программного обеспечения для компьютерных систем

*(наименование профессионального модуля)*

|  |  |  |
| --- | --- | --- |
| **№ п/п** | **Темы учебной практики** | **Выполненные задания/работы** |
| 1 | Тема 1. Программирование разветвляющегося процесса | Разработать программу сравнения двух чисел соблюдая следующие этапы:  - создать алгоритм;  - написать программный код в соответствии с разработанными алгоритмами и проектом;  - выполнить отладку;  - разработать техническую документацию. |
| 2 | Тема 2. Программирование разветвляющихся алгоритмов с помощью команд передачи управления языка Assembler | Реализовать алгоритм ветвления на языке Assembler соблюдая следующие этапы:  - создать алгоритм;  - написать программный код в соответствии с разработанными алгоритмами и проектом;  - выполнить отладку;  - разработать техническую документацию. |
| 3 | Тема 3. Работа с флагами регистра флагов с использованием команд языка Assembler | Разработать программу с использованием регистра FLAGS соблюдая следующие этапы:  - создать алгоритм;  - написать программный код в соответствии с разработанными алгоритмами и проектом;  - выполнить отладку;  - разработать техническую документацию. |
| 4 | Тема 4. Программирование алгоритмов с использованием арифметических команд и флага переноса языка Assembler | Разработать программу решения квадратного уравнения соблюдая следующие этапы:  - создать алгоритм;  - написать программный код в соответствии с разработанными алгоритмами и проектом;  - выполнить отладку;  - разработать техническую документацию. |
| 5 | Тема 5. Выполнение операций сдвига и логических операций над числами с помощью команд языка Assembler | Разработать программу с логических преобразований чисел соблюдая следующие этапы:  - создать алгоритм;  - написать программный код в соответствии с разработанными алгоритмами и проектом;  - выполнить отладку;  - разработать техническую документацию. |
| 6 | Тема 6. Умножение знаковых чисел | Разработать программу вычисления произведений произвольных чисел соблюдая следующие этапы:  - создать алгоритм;  - написать программный код в соответствии с разработанными алгоритмами и проектом;  - выполнить отладку;  - разработать техническую документацию. |
| 7 | Тема 7. Работа с флагами CF, OF. Деление знаковых чисел | Разработать программу вычисления деления произвольных чисел соблюдая следующие этапы:  - создать алгоритм;  - написать программный код в соответствии с разработанными алгоритмами и проектом;  - выполнить отладку;  - разработать техническую документацию. |
| 8 | Тема 8. Циклы | Разработать программу вычисления значений функций разложением в ряд Тейлора соблюдая следующие этапы:  - создать алгоритм;  - написать программный код в соответствии с разработанными алгоритмами и проектом;  - выполнить отладку;  - разработать техническую документацию. |
| 9 | Тема 9. Работа с сегментами. Подпрограммы | Разработать программу с использованием подпрограмм соблюдая следующие этапы:  - создать алгоритм;  - написать программный код в соответствии с разработанными алгоритмами и проектом;  - выполнить отладку;  - разработать техническую документацию. |
| 10 | Тема 10. Обработка одномерных массивов с использованием команд языка Assembler | Разработать программу обработки одномерного массива соблюдая следующие этапы:  - создать алгоритм;  - написать программный код в соответствии с разработанными алгоритмами и проектом;  - выполнить отладку;  - разработать техническую документацию. |
| 11 | Тема 11. Работа со стеком | Разработать программу с использованием стека соблюдая следующие этапы:  - создать алгоритм;  - написать программный код в соответствии с разработанными алгоритмами и проектом;  - выполнить отладку;  - разработать техническую документацию. |
| 12 | Тема 12. Работа с файлами | Разработать программу работы с файлом соблюдая следующие этапы:  - создать алгоритм;  - написать программный код в соответствии с разработанными алгоритмами и проектом;  - выполнить отладку;  - разработать техническую документацию. |
| 13 | Тема 13. Ассемблерная вставка в С++ | Разработать программу на языке соблюдая следующие этапы:  - определить спецификации;  - создать алгоритм;  - написать программный код в соответствии с разработанными алгоритмами и проектом;  - выполнить отладку;  - разработать техническую документацию. |
| 14 | Тема 14. Разработка программы с использованием Windows Forms | Разработать программу с использованием Windows Forms соблюдая следующие этапы:  - определить спецификации;  - создать алгоритм;  - написать программный код в соответствии с разработанными алгоритмами и проектом;  - выполнить отладку;  - разработать техническую документацию. |
| 15 | Тема 15. Написание программы для произведения стереометрических расчетов | Разработать программу для произведения стереометрических расчетов соблюдая следующие этапы:  - определить спецификации;  - создать алгоритм;  - написать программный код в соответствии с разработанными алгоритмами и проектом;  - выполнить отладку;  - разработать техническую документацию. |
| 16 | Тема 16. Написание консольного приложения для вывода числового массива | Разработать консольное приложение для вывода числового массива соблюдая следующие этапы:  - определить спецификации;  - создать алгоритм;  - написать программный код в соответствии с разработанными алгоритмами и проектом;  - выполнить отладку;  - разработать техническую документацию. |
| 17 | Тема 17. Написание приложения для работы с символьным массивом | Разработать приложение для работы с символьным массивом соблюдая следующие этапы:  - определить спецификации;  - создать алгоритм;  - написать программный код в соответствии с разработанными алгоритмами и проектом;  - выполнить отладку;  - разработать техническую документацию. |
| 18 | Тема 18. Создание приложения с использованием компонента ListBox | Разработать приложение с использованием компонента ListBox соблюдая следующие этапы:  - определить спецификации;  - создать алгоритм;  - написать программный код в соответствии с разработанными алгоритмами и проектом;  - выполнить отладку;  - разработать техническую документацию. |
| 19 | Тема 19. Создание приложения с использованием компонента DataGridView и многострочным textBox | Разработать приложение с использованием компонента DataGridView и многострочным textBox соблюдая следующие этапы:  - определить спецификации;  - создать алгоритм;  - написать программный код в соответствии с разработанными алгоритмами и проектом;  - выполнить отладку;  - разработать техническую документацию. |
| 20 | Тема 20. Создание приложения с для произведения планиметрических расчетов с использованием классов | Разработать приложение с для произведения планиметрических расчетов с использованием классов соблюдая следующие этапы: - определить спецификации;  - создать алгоритм; - при необходимости провести проектирование; - написать программный код в соответствии с разработанными алгоритмами и проектом; - выполнить отладку; - разработать техническую документацию. |
| 21 | Тема 21. Рефакторинг и оптимизация ранее разработанного приложения для обработки числового массива с использованием классов | Произвести разработку приложения для обработки числового массива с использованием классов соблюдая следующие этапы: - определить спецификации;  - создать алгоритм; - при необходимости провести проектирование; - написать программный код в соответствии с разработанными алгоритмами и проектом; - выполнить отладку; - разработать техническую документацию. |
| 22 | Тема 22. Создание приложения с использованием свойства и методов класса String | Разработать приложение с использованием свойств и методов класса String соблюдая следующие этапы: - определить спецификации;  - создать алгоритм; - при необходимости провести проектирование; - написать программный код в соответствии с разработанными алгоритмами и проектом; - выполнить отладку; - разработать техническую документацию. |
| 23 | Тема 23. Создание приложения с использованием делегатов (часть 1) | Разработать приложение с использованием делегатов соблюдая следующие этапы: - определить спецификации;  - создать алгоритм. |
| 24 | Тема 24. Создание приложения с использованием делегатов (часть 2) | Разработать приложение с использованием делегатов соблюдая следующие этапы:  - при необходимости провести проектирование; - написать программный код в соответствии с разработанными алгоритмами и проектом; - выполнить отладку; - разработать техническую документацию. |
| 25 | Тема 25. Создание приложения с использованием событий | Разработать приложение с использованием событий соблюдая следующие этапы: - определить спецификации;  - создать алгоритм; - при необходимости провести проектирование; - написать программный код в соответствии с разработанными алгоритмами и проектом; - выполнить отладку; - разработать техническую документацию. |
| 26 | Тема 26. Создание приложения «Калькулятор» с использованием классов | Разработать приложение «Калькулятор» с использованием событий соблюдая следующие этапы: - определить спецификации;  - создать алгоритм; - при необходимости провести проектирование; - написать программный код в соответствии с разработанными алгоритмами и проектом; - выполнить отладку; - разработать техническую документацию. |
| 27 | Тема 27. Создание приложения для проверки орфографии | Разработать приложение для проверки орфографии соблюдая следующие этапы: - определить спецификации;  - создать алгоритм; - при необходимости провести проектирование; - написать программный код в соответствии с разработанными алгоритмами и проектом; - выполнить отладку; - разработать техническую документацию. |
| 28 | Тема 28. Создание приложения для организации взаимодействия с MS Word | Разработать приложение для организации взаимодействия с MS Word соблюдая следующие этапы: - определить спецификации;  - создать алгоритм; - при необходимости провести проектирование; - написать программный код в соответствии с разработанными алгоритмами и проектом; - выполнить отладку; - разработать техническую документацию. |
| 29 | Тема 29. Создание приложения с использованием функций MS Excel | Разработать приложение с использованием функций MS Excel соблюдая следующие этапы:  - определить спецификации;  - создать алгоритм; - при необходимости провести проектирование; - написать программный код в соответствии с разработанными алгоритмами и проектом; - выполнить отладку; - разработать техническую документацию. |
| 30 | Тема 30. Создание поточного приложения | Разработать поточное приложение соблюдая следующие этапы: - определить спецификации;  - создать алгоритм; - при необходимости провести проектирование; - написать программный код в соответствии с разработанными алгоритмами и проектом; - выполнить отладку; - разработать техническую документацию. |
| 31 | Тема 31. Защита отчета по практике | Защита сформированного отчета по практике. |

|  |  |  |
| --- | --- | --- |
| **№п/п** | **Темы учебной практики** | **Выполненные задания/работы** |
| **1** | **Локальные данные процедур** | Дано описание: X DD 100 DUP(?) ; числа со знаком. Записать в регистр EAX число нулевых элементов массива X. |
| **2** | **Процедуры** | Ввести одномерные массивы a(5) и b(5), состоящие из произвольных чисел и определить в каком из массивов больше положительных элементов. Получить и вывести новый массив с, состоящий из положительных элементов массивов a и b. Программу разработать на основе пользовательских процедур (функций). |
| **3** | **Рекурсивные процедуры** | 1.Разработать рекурсивную процедуру вычисления суммы чисел от 1  до 100  2. Разработать рекурсивную процедуру вычисления суммы элементов массива X. |
| **4** | **Обработка двумерных массивов** | Разместить в памяти компьютера матрицу 4\*3 целых чисел со знаком. Написать программу, позволяющую увеличить на 10 отрицательные элементы массива. Исходную матрицу и полученную матрицу вывести на экран, |
| **5** | **Обработка одномерных массивов** | Максимальное отрицательное число |

**СОДЕРЖАНИЕ**

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 1**

**Тема: Локальные данные процедур**

Студент: Афонин Г.Д.

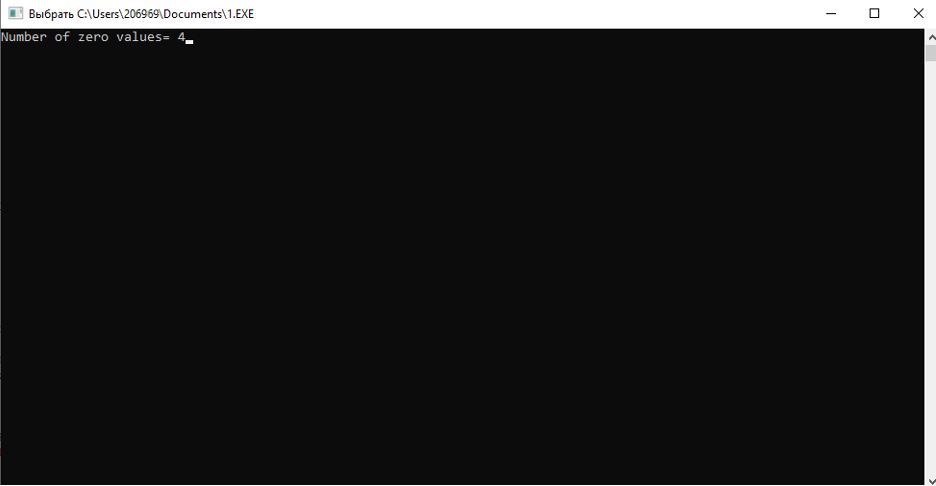
Группа: 3ПКС-420

Преподаватель: Альшакова Е.Л.

Дата:19.06.2022

**Цель работы**: Дано описание: X DD 100 DUP(?) ; числа со знаком. Записать в регистр EAX число нулевых элементов массива X.

**Ниже на рисунке представлен фрагмент работающего проекта**



**Код программы:**

format PE console

include 'win32a.inc'

start:

proc func

mov ecx, 15

lea esi, [X]

mov ebx, 0

mov eax,0

search1:

cmp [esi], ebx

jl next

cmp [esi],ebx

jg next

compare:

inc eax

next:

add esi, 4

loop search1

cinvoke printf, text1, eax

invoke sleep, 9000

invoke exit,0

ret

endp

call func

invoke sleep, 9000

invoke exit, 0

X dd 0, 2, 5, 7, 5, 3, 7, 9, 6, 1,4,0,0,5,0

;X dd 100 dup (?)

text db '', 10, 13, 0

text1 db 'Number of zero values= %d',0

fmat db '%d',9,0

; import data in the same section

data import

library msvcrt,'MSVCRT.DLL',\

kernel32,'KERNEL32.DLL'

import kernel32,\

sleep,'Sleep'

import msvcrt,\

printf,'printf',\

scanf,'scanf',\

exit,'exit',\

setlocale,'setlocale',\

system,'system'

end data

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 2**

**Тема: Процедуры**

Студент: Афонин Г.Д.

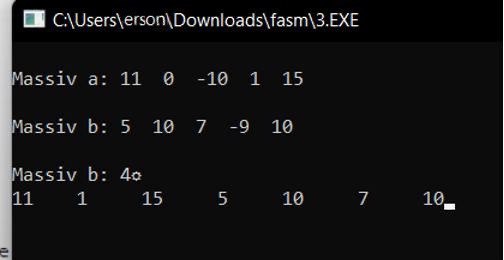
Группа: 3ПКС-420

Преподаватель: Альшакова Е.Л.

Дата:19.06.2022

**Цель работы**: Ввести одномерные массивы a(5) и b(5), состоящие из  
произвольных чисел и определить в каком из массивов больше  
положительных элементов. Получить и вывести новый массив с, состоящий  
из положительных элементов массивов a и b. Программу разработать на  
основе пользовательских процедур (функций).

**Ниже на рисунке представлен фрагмент работающего проекта**



**Код программы:**

format PE console

include 'win32a.inc'

start:

macro prnt2d op1,op2,op3

{

local met1,met2

mov esi,op1

mov ecx,op2

met2: push ecx

mov ecx,op3

met1: push ecx

mov eax,[esi]

push ecx

cinvoke printf,fmat,eax

pop ecx

add esi,4

pop ecx

loop met1

cinvoke printf, text3

pop ecx

loop met2

}

macro prntn op1,op2

{

local metka

mov esi,op1

mov ecx,op2

mov eax,0

metka:

mov eax,[esi]

push ecx

cinvoke printf,fmat,eax

pop ecx

add esi,4

loop metka

}

prnt2d X,[M],[N]

lea ebx,[Y]

lea esi,[X]

mov ecx,[M]

met2: push ecx

mov ecx,[N]

xor eax,eax

met1: add eax,[esi]

add esi,4

loop met1

mov [ebx],eax

add ebx,4

pop ecx

loop met2

xor eax,eax

lea ebx,[Y]

mov ecx,[M]

met3:

add eax,[ebx]

add ebx,4

loop met3

cinvoke printf, spp,eax

cinvoke printf, text4

prnt2d Y,1,[M]

invoke sleep,60000

invoke exit,0

X dd -1, -2, -3, 2, -1

dd 0, -1, -18, 15 -10

dd -1, -2, -3, 2, -1

dd 1, -1, 18, 10, -10

Y dd 100 dup(?)

N dd 5

M dd 4

text db 10,13, '', 10, 13, 0

text2 db 'Enter N: ',0

text3 db '',10,13,0

text4 db 10,13, '',10,13,0

fmat db '%d',9,0

spp db 10,13,'%d',0

data import

library msvcrt, 'MSVCRT.DLL',\

kernel132, 'KERNEL32.DLL'

import kernel132,\

sleep, 'Sleep'

import msvcrt,\

printf, 'printf',\

scanf,'scanf',\

exit, 'exit',\

setlocal, 'setlocal',\

system, 'system'

end data

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 3**

**Тема: Рекурсивные процедуры**

Студент: Афонин Г.Д.

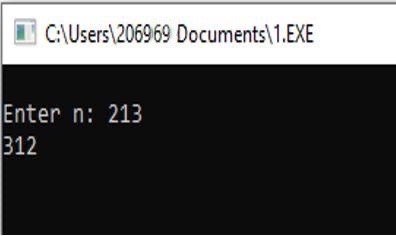
Группа: 3ПКС-420

Преподаватель: Альшакова Е.Л.

Дата:19.06.2022

**Цель работы**: Разработать рекурсивную процедуру вычисления суммы элементов массива X.

**Ниже на рисунке представлен фрагмент работающего проекта**



**Код программы:**

format PE console

include 'win32a.inc'

entry start

proc summa

mov ebx,[esi]

cmp ecx, dword 0

jg f1

ret

f1:

add eax, ebx

dec ecx

add esi,4

call summa

ret

endp

start:

mov eax,0

mov ecx,5

lea esi,[x]

call summa

cinvoke printf,text1,eax

cinvoke sleep,60000

cinvoke sleep,0

x dd 10,13,4,2,3

text1 db 10,13,'Summa(0-100): %d',0

text2 db '%d',0

fmat db ' %d',0

n dd ?

sum dd ?

data import

library msvcrt,'MSVCRT.DLL',\

kernel32,'KERNEL32.DLL'

import kernel32,\

sleep,'Sleep'

import msvcrt,\

printf,'printf',\

scanf,'scanf',\

exit,'exit',\

setlocale,'setlocale',\

system,'system'

end data

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 4**

**Тема: Обработка двумерных массивов**

Студент: Афонин Г.Д.

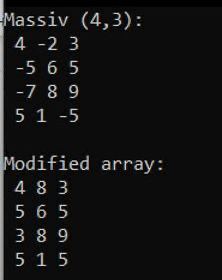
Группа: 3ПКС-420

Преподаватель: Альшакова Е.Л.

Дата:19.06.2022

**Цель работы**: Разработать процедуру обработки двумерного массива

**Ниже на рисунке представлен фрагмент работающего проекта**



**Код программы:**

format PE console

include 'win32a.inc'

entry start

proc change

mov ecx,[lines]

m:

push ecx

mov ecx,[columns]

mov eax,0

me:

cmp eax,[esi]

jg me1

jmp m1

me1:

mov ebx,[esi]

add ebx, 10

mov [esi],ebx

m1:

add esi,4

loop me

pop ecx

loop m

ret

endp

start:

macro prntn massiv,line,column

{

local metka

local met

mov esi,massiv

mov ecx,[lines]

met:

push ecx

mov ecx,[columns]

metka:

mov eax,[esi]

push ecx

cinvoke printf,fmat,eax

pop ecx

add esi,4

loop metka

cinvoke printf,text3

pop ecx

loop met

}

cinvoke printf,text1,[lines],[columns]

prntn mas,lines,columns

mov esi,mas

call change

cinvoke printf,text4

prntn mas,lines,columns

invoke sleep,5000

text1 db 'Massiv (%d,%d):',10,0

text3 db 10,13,'', 0

text4 db 10,13,'Modified array: ',10,13, 0

fmat db ' %d',0

p db 'pause', 0

min dd ?

ind dd 1

lines dd 4

columns dd 3

counter dd 1

mas dd 4,-2,3,-5,6,5,-7,8,9,5,1,-5

data import

library msvcrt, 'MSVCRT.DLL',\

kernel132, 'KERNEL32.DLL'

import kernel132,\

sleep, 'Sleep'

import msvcrt,\

printf, 'printf',\

scanf,'scanf',\

exit, 'exit',\

setlocal, 'setlocal',\

system, 'system'

end data

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 5**

**Тема: Обработка одномерных массивов**

Студент: Афонин Г.Д.

Группа: 3ПКС-420

Преподаватель: Альшакова Е.Л.

Дата:19.06.2022

**Цель работы**: Разработать процедуру обработки одномерного массива

**Код программы:**

5. format PE console

include 'win32a.inc'

entry start

proc negmax

mov ebx,esi

mov ecx,10

cycl:

cmp [esi],dword 0

jl met1

jmp met3

met1:

cmp ebx,[esi]

jg met2

jmp met3

met2:

mov ebx,esi

met3:

add esi,4

dec ecx

loop cycl

ret

endp

start:

macro prntn op1,op2

{

local metka

mov esi,op1

mov ecx,[op2]

mov eax,0

metka:

mov eax,[esi]

push ecx

cinvoke printf,fmat,eax

pop ecx

add esi,4

loop metka

}

prntn X,N

lea esi,[X]

call negmax

cinvoke printf,text1,[ebx]

invoke sleep,60000

invoke exit,0

X dd -6, -2, -3, 2, -1, 7, 3, 9,-16,10

N dd 10

text1 db 10,13,'NegMax: %d',0

fmat db '%d',9,0

spp db 10,13,'%d',0

data import

library msvcrt, 'MSVCRT.DLL',\

kernel132, 'KERNEL32.DLL'

import kernel132,\

sleep, 'Sleep'

import msvcrt,\

printf, 'printf',\

scanf,'scanf',\

exit, 'exit',\

setlocal, 'setlocal',\

system, 'system'

end data

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 6**

**Тема: Обработка выводных данных из консоли**

Студент: Афонин Г.Д.

Группа: 3ПКС-420

Преподаватель: Киреева Г.И.

Дата:19.06.2022

**Цель работы**: Требуется в консольном проекте создать двумерный числовой массив, заполнить его случайными целыми числами в диапазоне от 10 до 99.

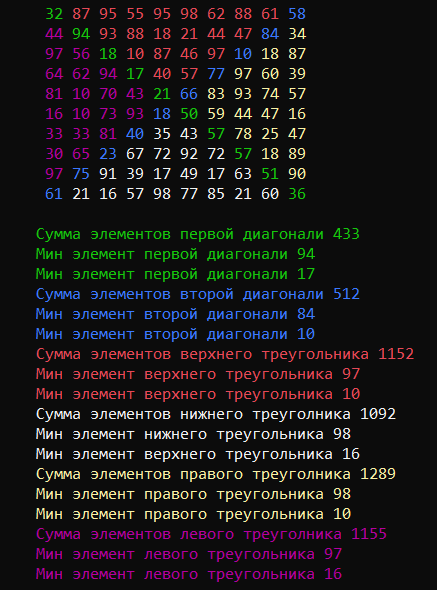
Вывести массив в консоль, установив разный цвет текста для следующих зон массива:

* Зоны главной и побочной диагоналей массива
* Зона выше главной и выше побочной диагоналей (верхний треугольник)
* Зона ниже главной и ниже побочной диагоналей (нижний треугольник)
* Зона выше главной и ниже побочной диагоналей (правый треугольник)
* Зона ниже главной и выше побочной диагоналей (левый треугольник).

Для каждой из зон вычислить следующие значения: sum, min, max, average.

Вывести в консоль эти значения с комментариями, используя для каждого из них такой же цвет текста, как и у соответствующей зоны массива

**Ниже на рисунке представлен фрагмент работающего проекта**



**Код программы:**

using System;

class Program

{

static void Main(string[] args)

{

int n = 10;

int k = 10;

var array = new int[n, k];

var random = new Random();

Console.ForegroundColor = ConsoleColor.Black;

Console.WriteLine();

raskraska(n, k);

Console.WriteLine();

}

/// <param name="n">Высота</param>

/// <param name="k">Длина</param>

static void raskraska(int n, int k)

{

var array = new int[n, k];

var random = new Random();

int sum1 = 0, sum2 = 0, sum3 = 0, sum4 = 0, sum5 = 0, sum6 = 0;

int max1 = 0, max2 = 0, max3 = 0, max4 = 0, max5 = 0, max6 = 0;

int min1 = 100, min2 = 100, min3 = 100, min4 = 100, min5 = 100, min6 = 100;

for (int i = 0; i < 10; i++)

{

Console.CursorLeft = 50;

for (int j = 0; j < 10; j++)

{

array[i, j] = random.Next(10, 100);

if (i == j)

{

if (array[i, j] > max1)

{

max1 = array[i, j];

}

if (array[i, j] < min1)

{

min1 = array[i, j];

}

sum1 += array[i, j];

Console.ForegroundColor = ConsoleColor.Green;

}

else if ((i + j) == (n - 1))

{

if (array[i, j] > max2)

{

max2 = array[i, j];

}

if (array[i, j] < min2)

{

min2 = array[i, j];

}

sum2 += array[i, j];

Console.ForegroundColor = ConsoleColor.Blue;

}

else if (i + j < n - 1 && i < j)

{

if (array[i, j] > max3)

{

max3 = array[i, j];

}

if (array[i, j] < min3)

{

min3 = array[i, j];

}

sum3 += array[i, j];

Console.ForegroundColor = ConsoleColor.Red;

}

else if ((i + j) > (n - 1) && i < j)

{

if (array[i, j] > max4)

{

max4 = array[i, j];

}

if (array[i, j] < min4)

{

min4 = array[i, j];

}

sum4 += array[i, j];

Console.ForegroundColor = ConsoleColor.Yellow;

}

else if ((i + j) > (n - 1) && i > j)

{

if (array[i, j] > max5)

{

max5 = array[i, j];

}

if (array[i, j] < min5)

{

min5 = array[i, j];

}

sum5 += array[i, j];

Console.ForegroundColor = ConsoleColor.White;

}

else if ((i + j) < (n - 1) && i > j)

{

if (array[i, j] > max6)

{

max6 = array[i, j];

}

if (array[i, j] < min6)

{

min6 = array[i, j];

}

sum6 += array[i, j];

Console.ForegroundColor = ConsoleColor.Magenta;

}

Console.Write(array[i, j] + " ");

}

Console.WriteLine();

}

Console.ForegroundColor = ConsoleColor.Green;

Console.WriteLine("\n\t\t\t\t\t\t Сумма элементов первой диагонали " + sum1);

Console.WriteLine("\t\t\t\t\t\t Мин элемент первой диагонали " + max1);

Console.WriteLine("\t\t\t\t\t\t Мин элемент первой диагонали " + min1);

Console.ForegroundColor = ConsoleColor.Blue;

Console.WriteLine("\t\t\t\t\t\t Сумма элементов второй диагонали " + sum2);

Console.WriteLine("\t\t\t\t\t\t Мин элемент второй диагонали " + max2);

Console.WriteLine("\t\t\t\t\t\t Мин элемент второй диагонали " + min2);

Console.ForegroundColor = ConsoleColor.Red;

Console.WriteLine("\t\t\t\t\t\t Сумма элементов верхнего треугольника " + sum6);

Console.WriteLine("\t\t\t\t\t\t Мин элемент верхнего треугольника " + max6);

Console.WriteLine("\t\t\t\t\t\t Мин элемент верхнего треугольника " + min6);

Console.ForegroundColor = ConsoleColor.White;

Console.WriteLine("\t\t\t\t\t\t Сумма элементов нижнего треуголника " + sum5);

Console.WriteLine("\t\t\t\t\t\t Мин элемент нижнего треугольника " + max5);

Console.WriteLine("\t\t\t\t\t\t Мин элемент верхнего треугольника " + min5);

Console.ForegroundColor = ConsoleColor.Yellow;

Console.WriteLine("\t\t\t\t\t\t Сумма элементов правого треуголника " + sum3);

Console.WriteLine("\t\t\t\t\t\t Мин элемент правого треугольника " + max3);

Console.WriteLine("\t\t\t\t\t\t Мин элемент правого треугольника " + min3);

Console.ForegroundColor = ConsoleColor.Magenta;

Console.WriteLine("\t\t\t\t\t\t Сумма элементов левого треуголника " + sum4);

Console.WriteLine("\t\t\t\t\t\t Мин элемент левого треугольника " + max4);

Console.WriteLine("\t\t\t\t\t\t Мин элемент левого треугольника " + min4);

Console.ResetColor();

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 7**

**Тема: Работа с WPF**

Студент: Афонин Г.Д.

Группа: 3ПКС-420

Преподаватель: Киреева Г.И.

Дата:19.06.2022

**Цель работы**: Создать форму вычислить функцию, под ней поле с функцией   
Изображение выглядит как текст

Автоматически созданное описание

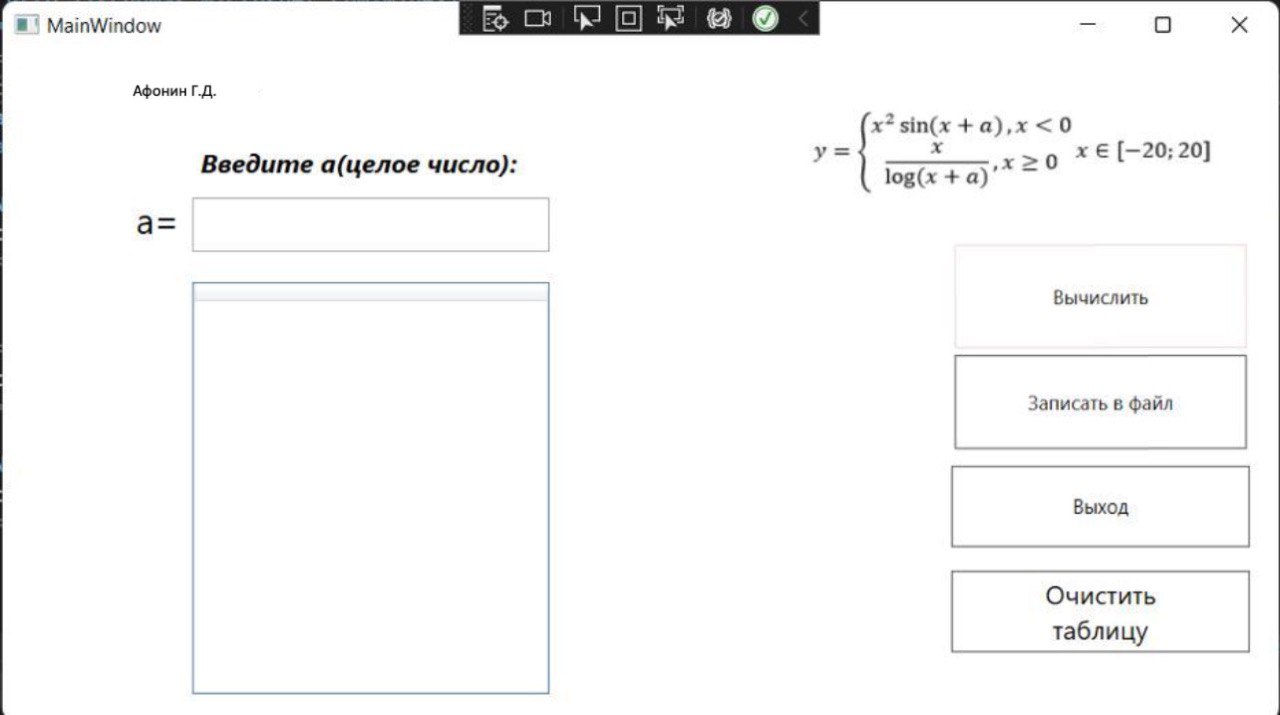
а вводится пользователем, при нажатии на кнопку вычислить – вычисляется функция

сетка грид под этим, в которой столбцы x | y со значениями

под кнопкой вычислить кнопка записать, она записывает в файл всю таблицу грид под кнопкой записать кнопка считать, она считывает из файла,

кнопка выхода.

**Ниже на рисунке представлен фрагмент работающего проекта**



**Код программы:**

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace WpfApp1

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

///

public class list

{

public int X { get; set; }

public string Y { get; set; }

public list(int x, string y)

{

X = x;

Y = y;

}

}

public partial class MainWindow : Window

{

public

string path = "note.exe";

public MainWindow()

{

InitializeComponent();

}

private void Exit\_Click(object sender, RoutedEventArgs e)

{

this.Close();

}

public void TextBox\_TextChanged(object sender, RoutedEventArgs e)

{

this.Show();

}

private void Form1\_Load(object sender, RoutedEventArgs e)

{

input.TabIndex = 0;

input.Focus();

}

double y;

List<list> result = new List<list>();

private void Calculate\_click(object sender, RoutedEventArgs e)

{

try

{

double.Parse(input.Text);

}

catch (FormatException)

{

MessageBox.Show("Введите число!");

return;

}

result.Clear();

Data.ItemsSource = null;

for (int i = -20; i <= 20; i++)

{

if (i >= 0)

{

if (i + double.Parse(input.Text) <= 0 || Math.Log10(i + double.Parse(input.Text)) <= 0)

{

result.Add(new list(i, "Не определён"));

continue;

}

else

{

double var = Math.Round(i / Math.Log10(i + double.Parse(input.Text)), 2);

result.Add(new list(i, var.ToString()));

}

}

else

{

double var = Math.Round(Math.Pow(i, 2) \* Math.Sin(i + double.Parse(input.Text)), 2);

result.Add(new list(i, var.ToString()));

}

}

input.Text = "";

Data.ColumnWidth = 97;

Data.ItemsSource = result;

}

private void Write\_Click(object sender, RoutedEventArgs e)//загрузить в файл

{

if (Data.ItemsSource == null) return;

StreamWriter sw = new StreamWriter("C:/Users/User/Documents/file.txt");

for (int i = 0; i <= 40; i++)

{

sw.Write(result[i].Y + "\n");

}

sw.Close();

MessageBox.Show("Запись успешно произведена!");

}

private void ClearGridData\_Click(object sender, RoutedEventArgs e)

{

Data.ItemsSource = null;

}

}

}

XAML:

<Window x:Class="WpfApp1.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:WpfApp1"

mc:Ignorable="d"

Title="MainWindow" Height="450" Width="800">

<Grid>

<Button Name="Exit\_click" Content="Выход" HorizontalAlignment="Left" Margin="584,256,0,0" VerticalAlignment="Top" Width="184" Height="50" Click="Exit\_Click" >

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Offset="0"/>

<GradientStop Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button Name="Calculate" Content="Вычислить" HorizontalAlignment="Left" Margin="586,120,0,0" VerticalAlignment="Top" Width="180" RenderTransformOrigin="0.5,0.5" Height="64" Click="Calculate\_click" BorderBrush="#FFEEDDDD">

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Offset="0"/>

<GradientStop Offset="0.487"/>

<GradientStop Offset="0.957"/>

</LinearGradientBrush>

</Button.Background>

<Button.RenderTransform>

<TransformGroup>

<ScaleTransform/>

<SkewTransform/>

<RotateTransform Angle="-0.053"/>

<TranslateTransform/>

</TransformGroup>

</Button.RenderTransform>

</Button>

<Button Name="ClearGridData" Content="Очистить&#xA; таблицу" HorizontalAlignment="Left" Margin="584,321,0,0" VerticalAlignment="Top" Width="184" Height="50" FontSize="16" Click="ClearGridData\_Click" >

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Offset="0"/>

<GradientStop Offset="1"/>

</LinearGradientBrush>

</Button.Background>

</Button>

<Button Name="Write" Content="Записать в файл" HorizontalAlignment="Left" Margin="586,188,0,0" VerticalAlignment="Top" Width="180" Height="58" Click="Write\_Click" >

<Button.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Offset="0"/>

<GradientStop Offset="0.487"/>

<GradientStop Offset="0.957"/>

</LinearGradientBrush>

</Button.Background>

<Button.RenderTransform>

<TransformGroup>

<ScaleTransform/>

<SkewTransform/>

<RotateTransform Angle="-0.053"/>

<TranslateTransform/>

</TransformGroup>

</Button.RenderTransform>

</Button>

<TextBox Name="input" HorizontalAlignment="Left" Height="34" Margin="117,91,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="220"/>

<TextBlock Margin="82,90,665,294" Text="a=" FontSize="22">

</TextBlock>

<DataGrid Name="Data" HorizontalAlignment="Left" Height="254" Margin="117,143,0,0" VerticalAlignment="Top" Width="220">

<DataGrid.Background>

<LinearGradientBrush EndPoint="0.5,1" StartPoint="0.5,0">

<GradientStop Offset="0"/>

<GradientStop Offset="0.487"/>

<GradientStop Offset="0.957"/>

</LinearGradientBrush>

</DataGrid.Background>

</DataGrid>

<Label Content="Введите a(целое число):" HorizontalAlignment="Left" VerticalAlignment="Top" Margin="117,54,0,0" FontSize="16" FontWeight="Bold" FontStyle="Italic" BorderBrush="#FFEEDBDB"/>

<Image Margin="487,33,24,323" Source="Image1.png" Stretch="Fill"/>

<Label Content="Афонин Г.Д." HorizontalAlignment="Left" Margin="61,10,0,0" VerticalAlignment="Top"/>

</Grid>

</Window>

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 8**

**Тема: Работа с ListBox**

Студент: Афонин Г.Д

Группа: 3ПКС-420

Преподаватель: Киреева Г.И.

Дата:19.06.2022

**Цель работы**:

- добавить окно ввода страны, обработать исключения(пробелы и тд)

-кнопка добавить, она добавляет в листбокс введённое предложение

-кнопка изменить, выбрать страну в листбоксе, при нажатии на кнопку открывается изменение этого слова с мигающим курсором

-кнопка удалить, также нужно выбрать и нажать кнопку, появится окно подтверждения с да/нет

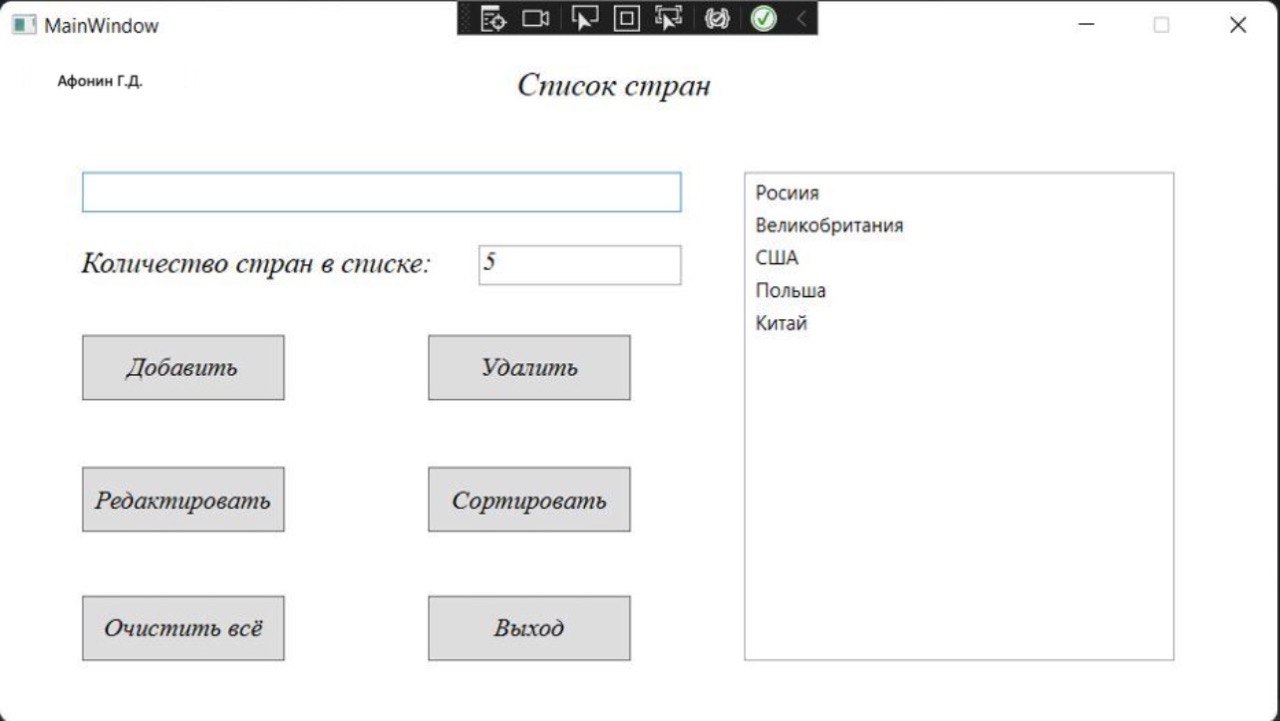
-очистить список, тоже самое, что и удаление, но предупреждение с восклицательным знаком

-сортировку она не показала, но думаю просто по алфавиту

-счетчик записей под окном ввода

-кнопка выхода

**Ниже на рисунке представлен фрагмент работающего проекта**

****

**Код программы:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace список

{

public partial class MainWindow : Window

{

int quantity\_products = 0; //количество продуктов

public MainWindow()

{

InitializeComponent();

input.Focus();

quantity.Text = quantity\_products.ToString();

}

private void click\_exit(object sender, RoutedEventArgs e)//выход из организации списка (с закрытием списка)

{

this.Close();

}

private void click\_add(object sender, RoutedEventArgs e)//добавление продукта

{

if (input.Text.Trim() == "")

{

MessageBox.Show("Вы ввели пустую строку!");

input.Focus();

}

else

{

quantity\_products += 1;

quantity.Text = quantity\_products.ToString();

list.Items.Add(input.Text.Trim());

input.Clear();

input.Focus();

}

}

private void click\_delete(object sender, RoutedEventArgs e)//удаление продукта

{

if (list.SelectedValue != null)

{

MessageBoxResult result = MessageBox.Show("Вы желаете удалить продукт из списка?", "Удалить", MessageBoxButton.YesNo, MessageBoxImage.Information);

if (result == MessageBoxResult.Yes)

{

list.Items.Remove(list.SelectedValue);

quantity\_products -= 1;

quantity.Text = quantity\_products.ToString();

if (quantity\_products < 0)

{

MessageBox.Show("Списк пуст, невозможно удаление продукта.");

quantity\_products = 0;

quantity.Text = quantity\_products.ToString();

}

input.Focus();

}

input.Focus();

}

else if (quantity\_products >= 0)

{

MessageBox.Show("Для удаления продукта из списка необходимо выделить его мышкой в окне списка.");

input.Focus();

}

else

{

MessageBox.Show("Списк пуст, невозможно удаление продукта.");

quantity\_products = 0;

quantity.Text = quantity\_products.ToString();

input.Focus();

}

}

private void click\_clear(object sender, RoutedEventArgs e)//очистка списка

{

MessageBoxResult result = MessageBox.Show("Вы желаете очистить весь список продуктов?", "Очистить всё", MessageBoxButton.YesNo, MessageBoxImage.Warning);

if (result == MessageBoxResult.Yes)

{

if (quantity\_products <= 0)

{

MessageBox.Show("Очищение пустого списка невозможно.");

input.Focus();

}

else

{

list.Items.Clear();

quantity\_products = 0;

quantity.Text = quantity\_products.ToString();

input.Focus();

}

}

}

private void click\_sort(object sender, RoutedEventArgs e)//сортировка

{

list.Items.SortDescriptions.Add(new System.ComponentModel.SortDescription("", System.ComponentModel.ListSortDirection.Ascending));

input.Focus();

}

private void click\_edit(object sender, RoutedEventArgs e)//редактирование

{

if (list.SelectedValue != null)

{

try

{

input.Text = list.SelectedValue.ToString();

list.Items.Remove(list.SelectedValue);

quantity\_products -= 1;

quantity.Text = quantity\_products.ToString();

}

catch { }

}

else

{

MessageBox.Show("Для редактирования списка продуктов необходимо выделить его мышкой в окне списка.");

input.Focus();

}

}

}

}

XAML:

<Window x:Class="список.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

ResizeMode="CanMinimize"

xmlns:local="clr-namespace:список"

mc:Ignorable="d"

Title="MainWindow" Height="450" Width="800">

<Grid>

<TextBlock HorizontalAlignment="Left" Height="28" Margin="318,10,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="140" FontSize="20" FontFamily="Times New Roman" FontStyle="Italic"><Run Text="Список "/><Run Language="ru-ru" Text="стран"/></TextBlock>

<ListBox Name ="list" HorizontalAlignment="Right" Height="300" Margin="0,75,63,0" VerticalAlignment="Top" Width="265"/>

<TextBox Name ="input" Height="25" Margin="50,75,0,0" TextWrapping="Wrap" Width="369"

HorizontalAlignment="Left" VerticalAlignment="Top"/>

<TextBox Name ="quantity" Height="25" Margin="294,120,0,0" TextWrapping="Wrap" Width="125" HorizontalAlignment="Left" VerticalAlignment="Top" FontFamily="Times New Roman" FontSize="16" FontStyle="Italic" TextDecorations="{x:Null}"/>

<Button Name ="buttton\_add" Click ="click\_add" Content="Добавить" HorizontalAlignment="Left" Height="40" Margin="50,175,0,0" VerticalAlignment="Top" Width="125" FontFamily="Times New Roman" FontSize="16" FontStyle="Italic"/>

<Button Name ="button\_delete" Click ="click\_delete" Content="Удалить" HorizontalAlignment="Left" Height="40" Margin="263,175,0,0" VerticalAlignment="Top" Width="125" FontFamily="Times New Roman" FontSize="16" FontStyle="Italic"/>

<Button Name ="button\_edit" Click ="click\_edit" Content="Редактировать" HorizontalAlignment="Left" Height="40" Margin="50,256,0,0" VerticalAlignment="Top" Width="125" FontFamily="Times New Roman" FontSize="16" FontStyle="Italic"/>

<Button Name ="button\_sort" Click ="click\_sort" Content="Сортировать" HorizontalAlignment="Left" Height="40" Margin="263,256,0,0" VerticalAlignment="Top" Width="125" FontFamily="Times New Roman" FontSize="16" FontStyle="Italic"/>

<Button Name ="button\_clear" Click ="click\_clear" Content="Очистить всё" HorizontalAlignment="Left" Height="40" Margin="50,335,0,0" VerticalAlignment="Top" Width="125" FontFamily="Times New Roman" FontSize="16" FontStyle="Italic" RenderTransformOrigin="0.498,0.256"/>

<Button Name ="button\_exit" Click ="click\_exit" Content="Выход" HorizontalAlignment="Left" Height="40" Margin="263,335,0,0" VerticalAlignment="Top" Width="125" FontFamily="Times New Roman" FontSize="16" FontStyle="Italic"/>

<TextBlock HorizontalAlignment="Center" Height="25" Margin="50,120,474.667,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="270" FontSize="18" FontFamily="Times New Roman" FontStyle="Italic"><Run Text=" Количество "/><Run Language="ru-ru" Text="стран"/><Run Text=" в списке: "/></TextBlock>

<TextBlock HorizontalAlignment="Left" Margin="22,10,0,0" TextWrapping="Wrap" VerticalAlignment="Top" RenderTransformOrigin="0.018,0.308"><Run Text="Афонин Г.Д."/><LineBreak/><Run/></TextBlock>

</Grid>

</Window>

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 9**

**Тема: Работа с БД**

Студент: Афонин Г.Д.

Группа: 3ПКС-420

Преподаватель: Киреева Г.И.

Дата:19.06.2022

**Цель работы**:

Создать базу данных «Торговое предприятие». В ней создать таблицы, которые относятся к этой базе. Установить связи между таблицами этой базы, заполнить их некоторыми данными. В отчёт вставить схемы данных и данные из таблиц.

**Ниже на рисунке представлен фрагмент работающего проекта**

Изображение выглядит как стол

Автоматически созданное описаниеИзображение выглядит как стол

Автоматически созданное описание

Изображение выглядит как стол

Автоматически созданное описаниеИзображение выглядит как стол

Автоматически созданное описание

Изображение выглядит как стол

Автоматически созданное описаниеИзображение выглядит как стол

Автоматически созданное описание

Изображение выглядит как стол

Автоматически созданное описание

Изображение выглядит как стол

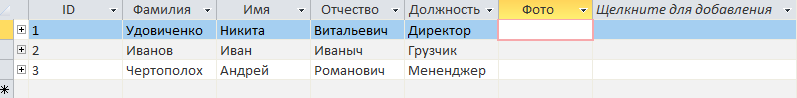
Автоматически созданное описание

Изображение выглядит как стол

Автоматически созданное описание

Изображение выглядит как стол

Автоматически созданное описание

Изображение выглядит как стол

Автоматически созданное описание

Изображение выглядит как стол

Автоматически созданное описание

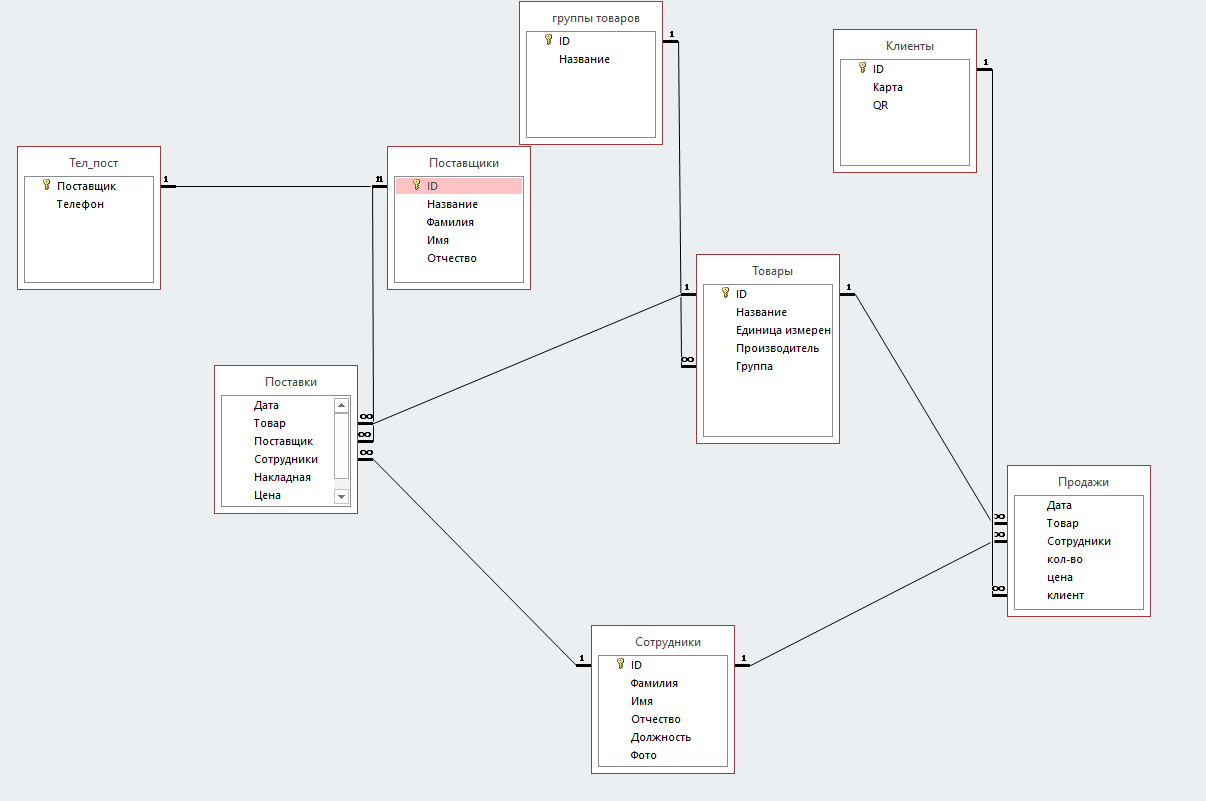


Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как стол

Автоматически созданное описание



Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 10**

**Тема: События C#”**

Студент: Афонин Г.Д.

Группа: 3ПКС-420

Преподаватель: Сибирев И.В.

Дата: 26.11.2022

**Цель работы**: разработать калькулятор на языке C#. В проекте использовать приём создания одного обработчика для нескольких событий.

**Ниже на рисунке представлен фрагмент работающего проекта**



**Код программы:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace WpfApp1

{

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

private void CE(object sender, RoutedEventArgs e)

{

TextBox.Clear();

}

private void N1\_Click(object sender, RoutedEventArgs e)

{

TextBox.Text += 1;

}

private void N2\_Click(object sender, RoutedEventArgs e)

{

TextBox.Text += 2;

}

private void N3\_Click(object sender, RoutedEventArgs e)

{

TextBox.Text += 3;

}

private void N4\_Click(object sender, RoutedEventArgs e)

{

TextBox.Text += 4;

}

private void N5\_Click(object sender, RoutedEventArgs e)

{

TextBox.Text += 5;

}

private void N6\_Click(object sender, RoutedEventArgs e)

{

TextBox.Text += 6;

}

private void N7\_Click(object sender, RoutedEventArgs e)

{

TextBox.Text += 7;

}

private void N8\_Click(object sender, RoutedEventArgs e)

{

TextBox.Text += 8;

}

private void N9\_Click(object sender, RoutedEventArgs e)

{

TextBox.Text += 9;

}

private void N0\_Click(object sender, RoutedEventArgs e)

{

TextBox.Text += 0;

}

private void Button\_Click(object sender, RoutedEventArgs e)

{

TextBox.Text += "√";

}

private void Button\_Click\_1(object sender, RoutedEventArgs e)

{

TextBox.Text += "/";

}

private void Button\_Click\_2(object sender, RoutedEventArgs e)

{

TextBox.Text += "\*";

}

private void Button\_Click\_3(object sender, RoutedEventArgs e)

{

TextBox.Text += "-";

}

private void Button\_Click\_4(object sender, RoutedEventArgs e)

{

TextBox.Text += "+";

}

private void ravno(object sender, RoutedEventArgs e)

{

TextBox.Text = Convert.ToString(Eval(TextBox.Text)).Replace(",", ".");

}

static Double Eval(String expression)

{

System.Data.DataTable table = new System.Data.DataTable();

return Convert.ToDouble(table.Compute(expression, String.Empty));

}

private void Button\_Click\_5(object sender, RoutedEventArgs e)

{

TextBox.Text += ".";

}

private void Button\_Click\_6(object sender, RoutedEventArgs e)

{

TextBox.Text = TextBox.Text.Remove(TextBox.Text.Length - 1);

}

}

}

XAML:

<Window x:Class="WpfApp1.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:WpfApp1"

mc:Ignorable="d"

Title="MainWindow" Background="Aquamarine" Height="360" Width="450">

<Grid Margin="0,0,-1,0" Background="Black">

<Grid.ColumnDefinitions>

<ColumnDefinition Width="277\*"/>

<ColumnDefinition Width="64\*"/>

<ColumnDefinition Width="81\*"/>

</Grid.ColumnDefinitions>

<Grid.RowDefinitions>

<RowDefinition Height="420\*"/>

<RowDefinition Height="11\*"/>

</Grid.RowDefinitions>

<TextBox Name="TextBox" HorizontalAlignment="Left" IsEnabled="False" Height="39" Margin="9,17,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="403" RenderTransformOrigin="0.496,0.433" Grid.ColumnSpan="3"/>

<Button Content="CE" HorizontalAlignment="Left" Margin="63.333,81,0,0" VerticalAlignment="Top" Width="72" Height="51" Click="CE" Grid.ColumnSpan="2" Grid.Column="1" />

<Button Name ="N1" Content="1" HorizontalAlignment="Left" Margin="9,200,0,0" VerticalAlignment="Top" Width="71" Height="51" Click="N1\_Click"/>

<Button Name ="N2" Content="2" HorizontalAlignment="Left" Margin="92,200,0,0" VerticalAlignment="Top" Width="71" Height="51" Click="N2\_Click"/>

<Button Name ="N3" Content="3" HorizontalAlignment="Left" Margin="175,200,0,0" VerticalAlignment="Top" Width="71" Height="51" Click="N3\_Click"/>

<Button Name ="N4" Content="4" HorizontalAlignment="Left" Margin="9,140,0,0" VerticalAlignment="Top" Width="71" Height="51" Click="N4\_Click"/>

<Button Name ="N5" Content="5" HorizontalAlignment="Left" Margin="92,140,0,0" VerticalAlignment="Top" Width="71" Height="51" Click="N5\_Click"/>

<Button Name ="N6" Content="6" HorizontalAlignment="Left" Margin="175,140,0,0" VerticalAlignment="Top" Width="71" Height="51" Click="N6\_Click"/>

<Button Name ="N7" Content="7" HorizontalAlignment="Left" Margin="9,81,0,0" VerticalAlignment="Top" Width="71" Height="51" Click="N7\_Click"/>

<Button Name ="N8" Content="8" HorizontalAlignment="Left" Margin="92,81,0,0" VerticalAlignment="Top" Width="71" Height="51" Click="N8\_Click"/>

<Button Name ="N9" Content="9" HorizontalAlignment="Left" Margin="175,81,0,0" VerticalAlignment="Top" Width="71" Height="51" Click="N9\_Click"/>

<Button Name ="N0" Content="0" HorizontalAlignment="Left" Margin="9,259,0,0" VerticalAlignment="Top" Width="154" Height="51" Click="N0\_Click"/>

<Button Content="." HorizontalAlignment="Left" Margin="175,259,0,0" VerticalAlignment="Top" Width="71" Height="51" Click="Button\_Click\_5"/>

<Button Content="/" HorizontalAlignment="Left" Margin="258,259,0,0" VerticalAlignment="Top" Width="70" Height="51" Click="Button\_Click\_1" Grid.ColumnSpan="2"/>

<Button Content="\*" HorizontalAlignment="Left" Margin="258,200,0,0" VerticalAlignment="Top" Width="70" Height="51" Click="Button\_Click\_2" Grid.ColumnSpan="2"/>

<Button Content="-" HorizontalAlignment="Left" Margin="258,140,0,0" VerticalAlignment="Top" Width="70" Height="51" Click="Button\_Click\_3" Grid.ColumnSpan="2"/>

<Button Content="+" HorizontalAlignment="Left" Margin="258,81,0,0" VerticalAlignment="Top" Width="70" Height="51" Click="Button\_Click\_4" Grid.ColumnSpan="2"/>

<Button Content="=" HorizontalAlignment="Left" Margin="63.333,200,0,0" VerticalAlignment="Top" Width="72" Height="110" Click="ravno" Grid.ColumnSpan="2" Grid.Column="1"/>

<Button Content="С" HorizontalAlignment="Left" Margin="63.333,140,0,0" VerticalAlignment="Top" Width="72" Height="51" Click="Button\_Click\_6" Grid.ColumnSpan="2" Grid.Column="1"/>

</Grid>

</Window>

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 11**

**Тема: “Checkspelling”**

Студент: Афонин Г.Д.

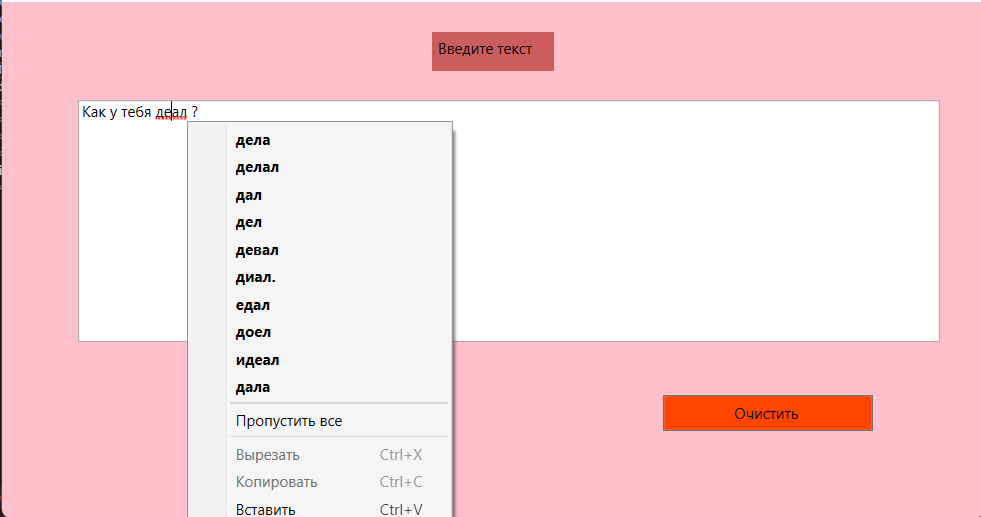
Группа: 3ПКС-420

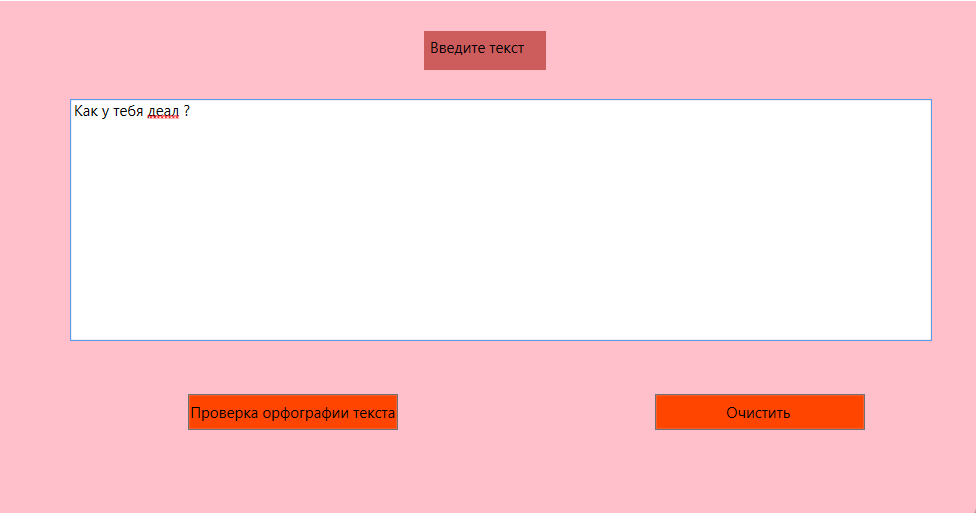
Преподаватель: Сибирев И.В.

Дата:26.11.2022

**Цель работы**: Разработать программу, которая предлагает пользователю ввести какие-либо слова, предложения в текстовое поле и после нажатия соответствующей кнопки проверить орфографию введенного текста. Для непосредственной проверки орфографии воспользуемся функцией Checkspelling объектной библиотеки MS Word.

**Ниже на рисунке представлен фрагмент работающего проекта**





**Код программы:**

using System;

using System.Collections.Generic;

using System.Windows;

using Word = Microsoft.Office.Interop.Word;

namespace Checkspelling

{

public partial class MainWindow

{

private void Button\_Click(object sender, RoutedEventArgs e)

{

int count = 0;

var Word = new Microsoft.Office.Interop.Word.Application();

var res = Word.CheckSpelling(TextBox.Text);

string[] mas = TextBox.Text.Split(' ');

foreach (string s in mas)

{

if (Word.CheckSpelling(s)==false)

{

count += 1;

MessageBox.Show("У вас ошибка в слове "+ s);

}

}

if (count == 0)

{

MessageBox.Show("Ваш текст полностью верный");

}

}

public MainWindow()

{

InitializeComponent();

}

private void Button\_Click\_1(object sender, RoutedEventArgs e)

{

TextBox.Clear();

}

}

}

XAML:

<Window x:Class="Checkspelling.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:Checkspelling"

mc:Ignorable="d"

Title="MainWindow" Height="450" Width="800" Background="Pink">

<Grid>

<Label Content="Введите текст" HorizontalAlignment="Center" Margin="0,24,0,0" Background="IndianRed" VerticalAlignment="Top" Height="31" Width="98"/>

<TextBox SpellCheck.IsEnabled="True" Name ="TextBox" HorizontalAlignment="Left" Margin="61,78,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="689" Height="194"/>

<Button Content="Проверка орфографии текста" HorizontalAlignment="Left" Background="OrangeRed" Margin="155,314,0,0" VerticalAlignment="Top" Height="29" Click="Button\_Click" />

<Button Content="Очистить " HorizontalAlignment="Left" Margin="529,314,0,0" Background="OrangeRed" VerticalAlignment="Top" Height="29" Width="168" Click="Button\_Click\_1"/>

</Grid>

</Window>

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 12**

**Тема: “ Создание приложения для организации взаимодействия с MS Word ”**

Студент: Афонин Г.Д.

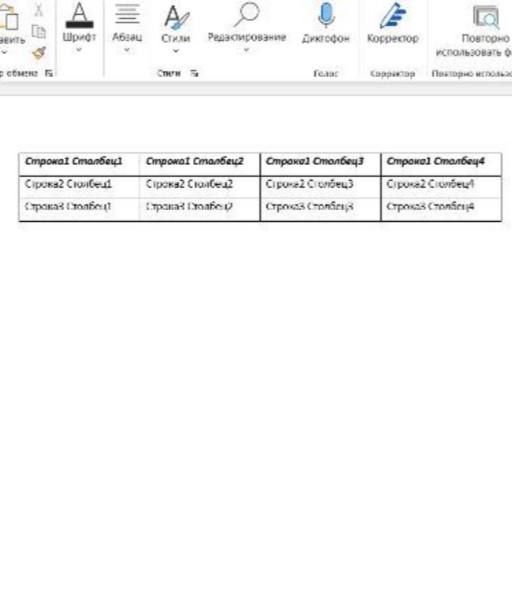
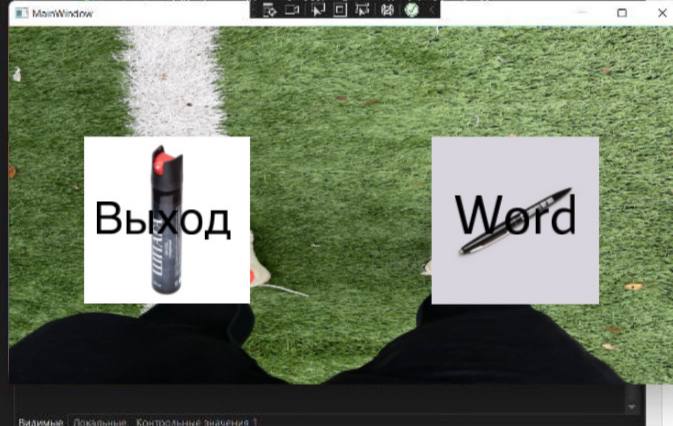
Группа: 3ПКС-420

Преподаватель: Сибирев И.В.

Дата:26.11.2022

**Цель работы**: необходимо разработать программу, которая при нажатии на кнопку создаёт таблицу в MS Word.

**Ниже на рисунке представлен фрагмент работающего проекта**



**Код программы:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

using System.Xml.Linq;

using Microsoft.Office.Interop.Word;

using Window = System.Windows.Window;

namespace Word\_Table

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

private void Exit(object sender, RoutedEventArgs e)

{

System.Windows.Application.Current.Shutdown();

}

private void Vord(object sender, RoutedEventArgs e)

{

createtable();

}

private void createtable()

{

object oMissing = System.Reflection.Missing.Value;

object oEndOfDoc = "\\endofdoc";

Microsoft.Office.Interop.Word.\_Application objWord;

Microsoft.Office.Interop.Word.\_Document objDoc;

objWord = new Microsoft.Office.Interop.Word.Application();

objWord.Visible = true;

objDoc = objWord.Documents.Add(ref oMissing, ref oMissing,

ref oMissing, ref oMissing);

int i = 0;

int j = 0;

Microsoft.Office.Interop.Word.Table objTable;

Microsoft.Office.Interop.Word.Range wrdRng = objDoc.Bookmarks.get\_Item(ref oEndOfDoc).Range;

objTable = objDoc.Tables.Add(wrdRng, 3, 4, ref oMissing, ref oMissing);

objTable.Range.ParagraphFormat.SpaceAfter = 7;

string strText;

for (i = 1; i <= 3; i++)

for (j = 1; j <= 4; j++)

{

strText = "Строка" + i + " Столбец" + j;

objTable.Cell(i, j).Range.Text = strText;

}

objTable.Rows[1].Range.Font.Bold = 1;

objTable.Rows[1].Range.Font.Italic = 1;

objTable.Borders.Shadow = true;

this.Close();

}

}

}

XAML:

<Window x:Class="Word\_Table.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:Word\_Table"

mc:Ignorable="d"

Title="MainWindow" Height="450" Width="800">

<Grid>

<Image Source="/Image1.png" Stretch="Fill" RenderTransformOrigin="0.505,0.535"/>

<Button HorizontalAlignment="Left" Margin="530,116,0,0" VerticalAlignment="Top" RenderTransformOrigin="0.609,0.413" Height="172" Width="214" Click="Vord">

<Image Source="/Resources/Гус с сигарой.jpg"></Image>

</Button>

<Button HorizontalAlignment="Left" Margin="85,116,0,0" VerticalAlignment="Top" Height="161" Width="172" Click ="Exit">

<Image Source="/Resources/Гус томас шелби.jpg" Height="170" Width="172" RenderTransformOrigin="0.5,0.5">

<Image.RenderTransform>

<TransformGroup>

<ScaleTransform/>

<SkewTransform AngleY="0.182"/>

<RotateTransform/>

<TranslateTransform Y="0.397"/>

</TransformGroup>

</Image.RenderTransform>

</Image>

</Button>

<Label Content="Word" HorizontalAlignment="Left" Margin="570,165,0,0" FontSize ="50" VerticalAlignment="Top" Height="86" Width="153"/>

<Label Content="Выход" HorizontalAlignment="Left" Margin="94,165,0,0" FontSize ="50" VerticalAlignment="Top" Height="76" Width="154"/>

</Grid>

</Window>

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 13**

**Тема: “Использование функций Excel”**

Студент: Афонин Г.Д.

Группа: 3ПКС-420

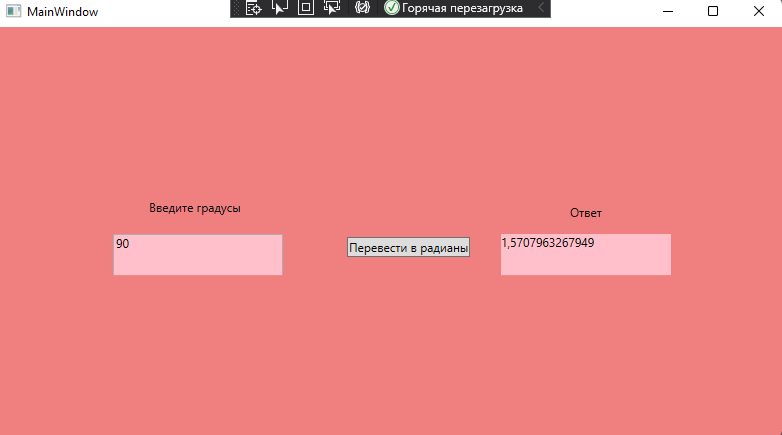
Преподаватель: Сибирев И.В.

Дата:26.11.2022

**Цель работы**: создать собственный проект с использованием функций MS

Excel не используя функцию нахождения числа Pi.

**Ниже на рисунке представлен фрагмент работающего проекта**



**Код программы:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace ExcelForm

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

private void ToRad(object sender, EventArgs e)

{

// Создание экземпляра класса Excel.Application:

var XL = new Microsoft.Office.Interop.Excel.Application();

var rad = XL.WorksheetFunction.Radians(Convert.ToDouble(TextBox.Text));

XL.Quit();

}

private void Button\_Click(object sender, RoutedEventArgs e)

{

// Создание экземпляра класса Excel.Application:

var XL = new Microsoft.Office.Interop.Excel.Application();

var rad = XL.WorksheetFunction.Radians(Convert.ToDouble(TextBox.Text));

TxtBlck.Text = Convert.ToString(rad);

XL.Quit();

}

}

}

XAML:

<Window x:Class="ExcelForm.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:ExcelForm"

mc:Ignorable="d"

Title="MainWindow" Height="450" Width="800" Background="LightCoral">

<Grid>

<Grid.RowDefinitions>

<RowDefinition Height="131\*"/>

<RowDefinition Height="86\*"/>

</Grid.RowDefinitions>

<TextBox Name="TextBox" Background="Pink" HorizontalAlignment="Left" Margin="115,207,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="170" Height="55"/>

<Label Content="Введите градусы&#xD;&#xA;" HorizontalAlignment="Left" Margin="146,167,0,0" VerticalAlignment="Top" Height="35" Width="108"/>

<Button Content="Перевести в радианы" HorizontalAlignment="Left" Margin="349,210,0,0" VerticalAlignment="Top" Click="Button\_Click" Height="20" Width="123"/>

<TextBlock Name="TxtBlck" Background="Pink" HorizontalAlignment="Left" Margin="503,207,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Height="55" Width="170"/>

<Label Content="Ответ" HorizontalAlignment="Left" Margin="567,172,0,0" VerticalAlignment="Top"/>

</Grid>

</Window>

Вывод: я разработал программу с использованием функции MS Excel, не используя функцию нахождения числа Pi

Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 14**

**Тема: “Текстовый редактор”**

Студент Афонин Г.Д.

Группа: 3ПКС-420

Преподаватель: Сибирев И.В.

Дата:25.11.2022

Москва, 2022

**Цель работы**: написать программу на языке C#, которая реализует

работу с файлами в различных кодировках.

Программа должна обладать следующим функционалом:

Чтение информации из файла в кодировке Unicode

Чтение информации из файла в кодировке Win1251

Чтение информации из файла с разрешением RTF

Чтение бинарных файлов

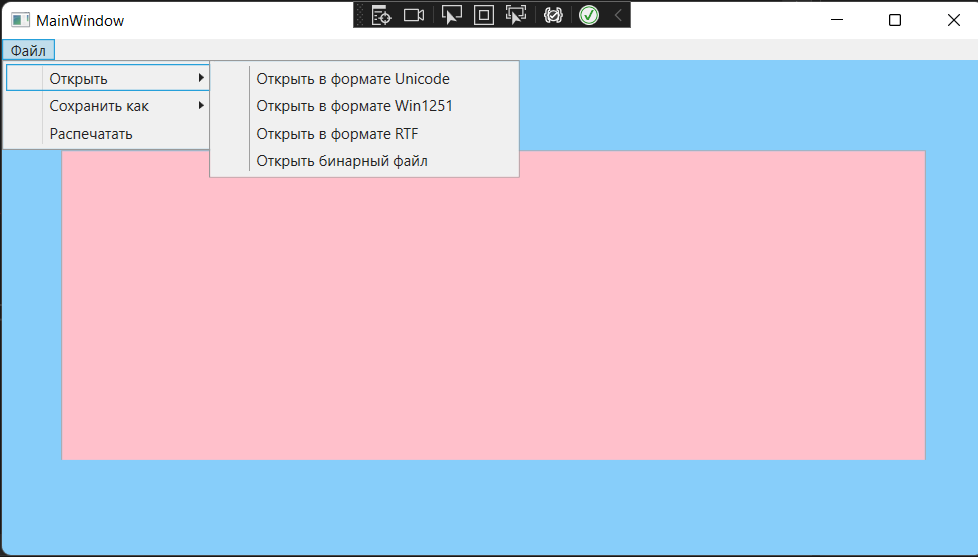
Сохранение информации в файл с кодировкой Unicode

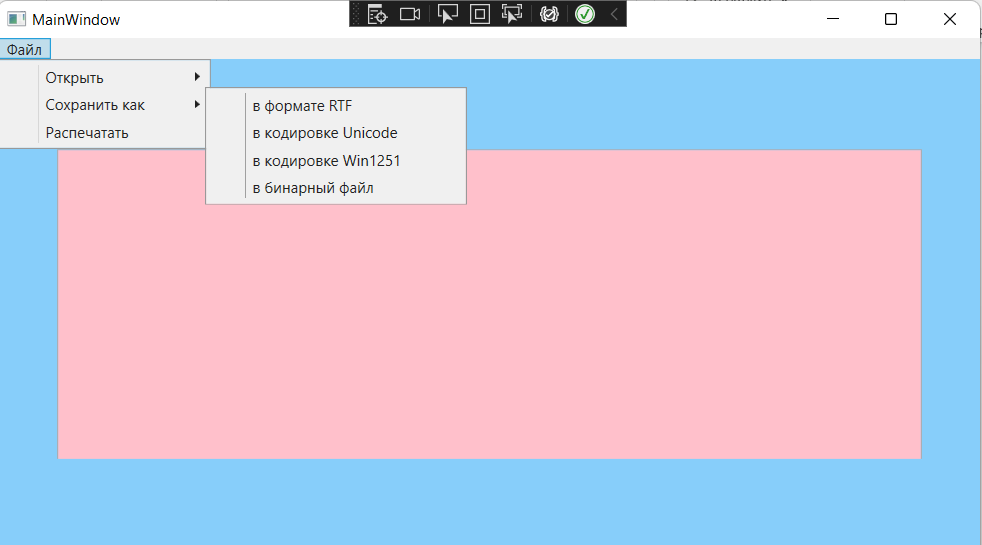
Сохранение информации в файл с кодировкай Win1251

Сохранение информации в файл с разрешением RTF

Сохранение информации в бинарный файл

Печать текстового документа.

**Ниже на рисунке представлен фрагмент работающего проекта**



**Код программы:**

using Microsoft.Win32;

using System;

using System.IO;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

using Microsoft.SqlServer.Server;

using System.Runtime.Serialization.Formatters.Binary;

namespace Text\_editor

{

public partial class MainWindow : Window

{

Microsoft.Win32.SaveFileDialog dlg = new Microsoft.Win32.SaveFileDialog();

Microsoft.Win32.OpenFileDialog odlg = new Microsoft.Win32.OpenFileDialog();

public MainWindow()

{

InitializeComponent();

dlg.FileName = "Документ";

odlg.FileName = "Документ";

}

private void Save\_as\_RTF(object sender, RoutedEventArgs e)

{

dlg.DefaultExt = ".rtf";

dlg.Filter = "Текстовый документ (.rtf)|\*.rtf";

string richText = new TextRange(rtxtbx.Document.ContentStart, rtxtbx.Document.ContentEnd).Text;

if (dlg.ShowDialog() == false)

{

return;

}

string filename = dlg.FileName;

File.WriteAllText(filename, richText);

MessageBox.Show("Файл успешно сохранён!");

}

private void Save\_as\_Unicode(object sender, RoutedEventArgs e)

{

dlg.DefaultExt = ".txt";

dlg.Filter = "Текстовый документ (.txt)|\*.txt";

string richText = new TextRange(rtxtbx.Document.ContentStart, rtxtbx.Document.ContentEnd).Text;

if (dlg.ShowDialog() == false)

{

return;

}

string filename = dlg.FileName;

File.WriteAllText(filename, richText, Encoding.Unicode);

MessageBox.Show("Файл успешно сохранён!");

}

private void Save\_as\_Win1251(object sender, RoutedEventArgs e)

{

dlg.DefaultExt = ".txt";

dlg.Filter = "Текстовый документ (.txt)|\*.txt";

string richText = new TextRange(rtxtbx.Document.ContentStart, rtxtbx.Document.ContentEnd).Text;

if (dlg.ShowDialog() == false)

{

return;

}

string filename = dlg.FileName;

File.WriteAllText(filename, richText, Encoding.GetEncoding("windows-1251"));

MessageBox.Show("Файл успешно сохранён!");

}

private void Save\_as\_BinaryFormat(object sender, RoutedEventArgs e)

{

dlg.DefaultExt = ".bin";

dlg.Filter = "Текстовый документ (.bin)|\*.bin";

BinaryFormatter bf = new BinaryFormatter();

if (dlg.ShowDialog() == false)

{

return;

}

string filename = dlg.FileName;

string richText = new TextRange(rtxtbx.Document.ContentStart, rtxtbx.Document.ContentEnd).Text;

using (BinaryWriter writer = new BinaryWriter(File.Open(filename, FileMode.OpenOrCreate)))

{

writer.Write(richText);

MessageBox.Show("Файл успешно сохранён!(на рабочий стол)");

}

}

private void Open\_as\_RTF(object sender, RoutedEventArgs e)

{

rtxtbx.Document.Blocks.Clear();

odlg.DefaultExt = ".rtf";

odlg.Filter = "Текстовый документ (.rtf)|\*.rtf";

bool? dialogResult = odlg.ShowDialog();

if (dialogResult.HasValue && dialogResult.Value)

{

TextRange range = new TextRange(rtxtbx.Document.ContentStart, rtxtbx.Document.ContentEnd);

System.IO.FileStream file = new System.IO.FileStream(odlg.FileName, System.IO.FileMode.Open, System.IO.FileAccess.Read);

range.Load(file, System.Windows.DataFormats.Rtf);

file.Dispose();

MessageBox.Show("Файл открыт!");

}

}

private void Open\_as\_Unicode(object sender, RoutedEventArgs e)

{

string filename = "";

rtxtbx.Document.Blocks.Clear();

bool? result = odlg.ShowDialog();

if (result == true)

{

filename = odlg.FileName;

}

string filetext = File.ReadAllText(filename);

rtxtbx.Document.Blocks.Add(new Paragraph(new Run(filetext)));

MessageBox.Show("Файл открыт!");

}

private void Open\_as\_Win1251(object sender, RoutedEventArgs e)

{

string filename = "";

rtxtbx.Document.Blocks.Clear();

bool? result = odlg.ShowDialog();

if (result == true)

{

filename = odlg.FileName;

}

string filetext = File.ReadAllText(filename, System.Text.Encoding.GetEncoding("windows-1251"));

rtxtbx.Document.Blocks.Add(new Paragraph(new Run(filetext)));

MessageBox.Show("Файл открыт!");

}

private void Open\_as\_Binary(object sender, RoutedEventArgs e)

{

string filename = "";

rtxtbx.Document.Blocks.Clear();

bool? result = odlg.ShowDialog();

if (result == true)

{

filename = odlg.FileName;

}

string filetext = File.ReadAllText(filename);

rtxtbx.Document.Blocks.Add(new Paragraph(new Run(filetext)));

MessageBox.Show("Файл открыт!");

}

private void PrintWindow(object sender, RoutedEventArgs e)

{

PrintDialog p = new PrintDialog();

if(p.ShowDialog() == true)

{

p.PrintVisual(grid1, "Печать");

}

}

}

}

XAML:

<Window x:Class="Text\_editor.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:Text\_editor"

mc:Ignorable="d"

Title="MainWindow" Height="450" Width="800" Background="LightSkyBlue">

<Grid Name="grid1">

<Grid.RowDefinitions>

<RowDefinition Height="177\*"/>

<RowDefinition Height="40\*"/>

</Grid.RowDefinitions>

<Menu Margin="0,0,0,320">

<MenuItem Header="Файл">

<MenuItem Header="Открыть">

<MenuItem Header="Открыть в формате Unicode" Click="Open\_as\_Unicode"></MenuItem>

<MenuItem Header="Открыть в формате Win1251" Click="Open\_as\_Win1251"></MenuItem>

<MenuItem Header="Открыть в формате RTF" Click="Open\_as\_RTF" ></MenuItem>

<MenuItem Header="Открыть бинарный файл" Click="Open\_as\_Binary"></MenuItem>

</MenuItem>

<MenuItem Header="Сохранить как">

<MenuItem Header="в формате RTF" Click="Save\_as\_RTF"></MenuItem>

<MenuItem Header="в кодировке Unicode" Click="Save\_as\_Unicode"></MenuItem>

<MenuItem Header="в кодировке Win1251" Click="Save\_as\_Win1251"></MenuItem>

<MenuItem Header="в бинарный файл" Click="Save\_as\_BinaryFormat"></MenuItem>

</MenuItem>

<MenuItem Header="Распечатать" Click="PrintWindow"></MenuItem>

</MenuItem>

</Menu>

<RichTextBox Name="rtxtbx" HorizontalAlignment="Center" Background="Pink" VerticalAlignment="Top" Width="692" Height="256" Margin="0,89,0,0">

<FlowDocument>

<Paragraph>

<Run/>

</Paragraph>

</FlowDocument>

</RichTextBox>

</Grid>

</Window>