Инв. № подл Подп. и дата Взам. инв. № Инв. № дубл. Подп. и дата

ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ «ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет компьютерных наук Департамент программной инженерии

СОГЛАСОВАНО	УТВЕРЖДАЮ
Преподаватель департамента программной	Академический руководитель
инженерии факультета компьютерных	образовательной программы
наук	«Программная инженерия» профессор
v	департамента программной
	инженерии, канд. техн. наук
Н. К. Чуйкин	В. В. Шилов
«» 2019 г.	«» 2019 г.
РАЗРАБОТКА ПРОГРАММ ВИЗУАЛИЗАЦИИ ГРАФА Д СЕТИ "ВКО	ДРУЗЕЙ СОЦИАЛЬНОЙ
Пояснительна	
ЛИСТ УТВЕР	РЖДЕНИЯ
$\mathrm{RU}.17701729.04.13$	-01 51 01-1-ЛУ
	Исполнитель: студент группы БПИ185
	А. А. Мануйлов
	« » 2019 г.

РАЗРАБОТКА ПРОГРАММНОГО ПРОДУКТА ДЛЯ ВИЗУАЛИЗАЦИИ ГРАФА ДРУЗЕЙ СОЦИАЛЬНОЙ СЕТИ "ВКОНТАКТЕ"

Пояснительная записка

лист утверждения

 $RU.17701729.04.13-01\ 51\ 01-1-ЛУ$

Листов 16

Содержание

1	Вве	дение	4
	1.1	Наименование программы	4
	1.2	Документы, на основании которых ведется разработка	4
2	Has	начение и область применения	5
	2.1	Назначение программы	5
		2.1.1 Функциональное назначение	5
		2.1.2 Эксплуатационное назначение	5
		2.1.3 Область применения	5
3	Tex	нические характеристики	6
	3.1	Постановка задачи на разработку программы	6
	3.2	Описание алгоритмов и функционирования программы	6
		3.2.1 Описание алгоритмов программы	6
		3.2.2 Возможные взаимодействия программы с другими программами.	7
	3.3	Описание и обоснование выбора метода организации входных и выходных	
		данных	7
		3.3.1 Описание метода организации входных и выходных данных	7
		3.3.2 Обоснование выбора метода организации входных и выходных дан-	
		ных	7
	3.4	Описание и обоснование выбора состава технических и программных средств	7
		3.4.1 Состав технических и программных средств	7
4	Tex	нико-экономические показатели	8
	4.1	Предполагаемая потребность	8
		4.1.1 Экономические преимущества по сравнению с отечественными и	
		зарубежными аналогами	8
Cı	писо	к источников	9
5	Прі	ложение: описание классов, полей и методов	LO
6	Диа	аграмма классов 1	L 5
7	Лис	т регистрации изменений	16

1 Введение

1.1 Наименование программы

Программа для визуализации графа друзей социальной сети "Вконтакте" Programm for Visualisation Graph of Friends in the Social Network "VK"

1.2 Документы, на основании которых ведется разработка

Приказ декана факультета компьютерных наук Национального Исследовательского университета «Высшая школа экономики» № 2.3-02/1012-0 2 «Об утверждении тем, руководителей курсовых работ студентов образовательной программы «Программная инженерия» факультета компьютерных наук» от 10.12.18

2 Назначение и область применения

2.1 Назначение программы

2.1.1 Функциональное назначение

Функциональным назначением программы является построение графа друзей социальной сети «ВКонтакте», демонстрация кластеризированности окружения пользователя, наглядное представление работы силового алгоритма.

2.1.2 Эксплуатационное назначение

Эксплуатационным назначением программы является предоставление пользователю возможности построить кластеризированный граф социальной сети своего аккаунта «ВКонтакте». Приложение может быть использованно для анализа кластерного разделения друзей «ВКонтакте».

2.1.3 Область применения

Данная программа применяется для визуализации структуры социального окружения пользователя и анализа взаимосвязей друзей.

3 Технические характеристики

3.1 Постановка задачи на разработку программы

Задача - разработка программы для построения графа друзей социальной сети «ВКонтакте», его дальнейшей кластеризации и приминения к нему силового алгоритма LPA. Пользователь имеет возможность выбрать вершину графа с ее последующим выделением и получить информацию о выбранной вершине (имя выбранного пользователя и фотография). Программы реализована на платформе Windows Forms Application на языке программирования С#.

3.2 Описание алгоритмов и функционирования программы

3.2.1 Описание алгоритмов программы

Программа содержит основные алгоритмы:

1. Силовой алгоритм

Основная идея алгорима заключается в проведении аналогии между графом на плоскости и физической системой. Цель силового алгоритма - расположение вершин графа максимально удобно для визуального восприятия, этой задаче прекрасно удовлетворяет ситуация минимальной энергии в системе. Связанные вершины притягиваются, как будто соединающие их ребра это пружины (проводится анология с законом Гука), а не связанные вершины отталкиваются, как будто это заряженные противоположными зарядами частицы (аналогия с законом Кулона). Кроме того, все вершины притягиваются к центру поля (вершине пользователя программы), для того, чтобы граф не "разлетался" в стороны.

При запуске алгоритма происходит фиксированное количество итераций (200). Во время каждой итерации вычисляется суммарный вектор силы, действующий на каждую вершину со стороны других и после все вершины сдвигаются на небольшое расстояние в направлении этих сил.

Основное отличие алгоритма от модели реальной физической системы это формулы вычисления сил. Сила притяжения считается по логарифмической зависимости:

 $F = \log_{10}(\frac{D}{C_2}) \cdot C_1$, где D - расстояние между вершинами, C_1 - константа нормировки, C_2 - идеальное расстояние между вершинами (1/20 диагонали экрана). Как видно из формулы, использование логарифма не дает вершинам "слипаться"и их притяжение становится более плавным, чем по линейному закону. Сила отталкивания считается по обратно квадратической зависимости от расстояния:

 $F = \frac{-C_3}{D^2}$, где C_3 - константа нормировки, D - расстояние между вершинами.

Притяжение к центру же просто пропорционально расстоянию от вершины до центра. 2. Алгоритм кластеризации LPA

Алгоритм распространения меток (Label Propagation Algorithm) является алгоритмом строгой кластеризации (каждая вершина принадлежит только одному кластеру). Этот алгоритм достаточно прост в реализации и в исполнении, и к тому же он достаточно хорошо имитирует знакомства людей в реальной жизни.

Суть алгоритма в том, что на старте каждой вершине присваивается уникальный идентификатор, а затем, пока изменения в метках вершин не прекратятся, происходит следущее: для каждой вершины считается самая часто встречаемая метка среди ее соседей, и затем эта метка присваивается самой вершине. Если таких меток несколько, то из них выбирается случайно одна. Таким образом, все вершины оказываются разделены на кластеры, которые в последствии удобно воспринимать.

3.2.2 Возможные взаимодействия программы с другими программами

Программа предусматривает взаимодействие с vk арі для авторизации пользователя и получения токена (ключа доступа), получения списка друзей пользователя, получения списка общих друзей и получения общей информации о пользователе.

3.3 Описание и обоснование выбора метода организации входных и выходных данных

3.3.1 Описание метода организации входных и выходных данных

Входные данные пользователь вводит через дополнительную форму в поля ввода пароля и логиня на странице «Вконтакте». Выходными данными является изображение графа в формате .JPEG

3.3.2 Обоснование выбора метода организации входных и выходных данных

Ввод данных в поля ввода «Вконтакте», минуя код программы, - самый безопасный способ авторизации.

3.4 Описание и обоснование выбора состава технических и программных средств

3.4.1 Состав технических и программных средств

Для надёжной и бесперебойной работы программы требуется следующий состав технических средств:

- 1) персональный компьютер, оснащенный 64-разрядным (x64) процессором Pentium с тактовой частотой 1.6 ГГц и выше или аналогичный процессор;
- 2) 4 ГБ оперативной памяти или больше;
- 3) не менее 1 ГБ свободного места на жестком диске;
- 4) видеокарта и монитор, поддерживающие режим Super VGA с разрешением не менее чем 850x600 точек;
- 5) мышь Microsoft Mouse или совместимое указывающее устройство;
- 6) клавиатура;

4 Технико-экономические показатели

4.1 Предполагаемая потребность

4.1.1 Экономические преимущества по сравнению с отечественными и зарубежными аналогами

По сравнению с существующими аналогами данная программа является десктопной, что позволяет пользоватся ей не прибегая к работе браузеров. Ненагруженный интерфейс и интуитивное понимание делают программу легко осваиваемой и простой в использовании даже для начинающего пользоваля.

Список источников

- [1] Vuslit [Электронный ресурс] URL: https://vuzlit.ru/949014/avtomaticheskoe_raspolozhenie_vershin_ploskosti (Дата обращения: 09.05.2019, режим доступа: свободный)
- [2] Полное руководство по языку программирования С# 7.0 и платформе .NET 4.7 Metanit [Электронный ресурс] URL: https://metanit.com/sharp/tutorial/ (Дата обращения: 21.05.2019, режим доступа: свободный)
- [3] Руководство по языку С# MSDN [Электронный ресурс] URL: https://docs.microsoft.com/ru-ru/dotnet/csharp/ (Дата обращения: 30.04.2019, режим доступа: свободный)
- [4] Руководство разработчикам по API ВКонтакте [Электронный ресурс] URL: https://vk.com/dev/first_guide (Дата обращения: 26.05.2019, режим доступа: свободный)
- [5] Герберт Шилдт. Полное руководство С# 4.0. 2014. -1056.

5 Приложение: описание классов, полей и методов

Таблица 1: Описание классов и их основного функционала

Класс	Функционал			
Form1	Основное окно приложения, содержит контрол PictureBox			
	для отрисовки графа, контрол PictureBox для отображения фотографии			
	профиля авторизовавшегося пользователя, контрол PictureBox для			
	отображения фотографии выбраного на графе пользователя, несколько			
	контролов Label для вывода информации о пользователях, контрол			
	menuStrip для выбора необходимой операции. Данный класс			
	содержит все основыные методы и события для работы программы.			
browser	Вспомогательная форма для авторизации пользователя,			
	получения и передачи в основную форму accessToken			
Vertex	Класс, представляющий вершину на плоскости			
Edge	Класс, Представляющий ребро на плоскости			

Таблица 2: Описание полей класса Edge

Имя	Модификатор доступа	Тип	Назначение
V1	public	Vertex	Первая вершина ребра
V2	public	Vertex	Вторая вершина ребра

Таблица 3: Описание методов класса Edge

Имя	Модификатор доступа	Тип	Аргументы	Назначение
IsVertexPicked	public	bool	-	Проверяет выбрана
				ли одна из вершин

Таблица 4: Описание полей класса Vertex

Имя	Модификатор доступа	Тип	Назначение
Coord	public	PointF	Координаты вершины

Таблица 5: Описание методов класса Vertex

Имя	Модификатор доступа	Тип	Аргументы	Назначение
IsVertexCommon	public	bool	Vertex v	Проверяет связаны
				ли вершины

Таблица 6: Описание свойств класса Vertex

Имя	Модификатор доступа	Тип	Доступ	Назначение
Common_Vertexes	public	string[]	get	Проверяет связаны
				ли вершины
ID	public	string	get	id вершины
				(пользователя)
Group	public	string	get set	Принадлежность кластеру
IsPicked	public	bool	get set	Выбрана ли вершина

Таблица 7: Описание полей класса browser

Имя	Модификатор доступа	Тип	Назначение
client_id	private	string	id приложения вконтакте
accessToken	public	string	токен доступа
userId	public	string	id авторизовавшегося пользователя
address	private	string	URL для перехода на
			страницу авторизации

Таблица 8: Описание методов класса browser

Имя	Модификатор доступа	Тип	Аргументы	Назначение
webBrowser1_	public	void	-	Событие обработчик
DocumentCompleted				загрузки страницы
				Получает из URL
				токен и id

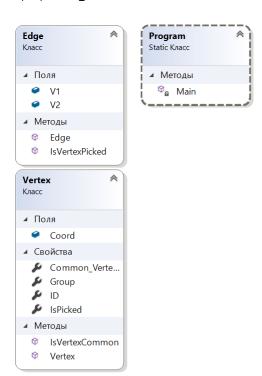
Таблица 9: Описание полей класса Form1

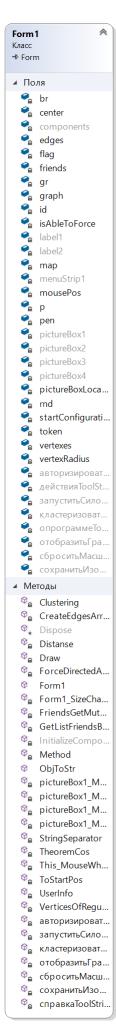
Имя	Marydywanan	Тип	Назначение
RMN	Модификатор	ТИП	пазначение
	доступа		
map	private	Bitmap	Битмап
gr	private	Graphics	Графика для рисования
br	private	SolidBrush	Кисть для рисования вершин
rnd	private	Random	Для произвольного расположения
mousePos	private	PointF	Временная точка для перерисовки
center	private	PointF	Координаты центральной точки
p	private	PointF	Временная точка для перерисовки
picture	private	Point	Расположение контрола
BoxLocation			
			pictureBox на форме во время его
			перемещения
vertexes	private	Vertex[]	Все вершины
edges	private	Edge[]	Все ребра
start	private	PointF[]	Точки в старт
Configuration			
pen	private	Pen	Ручка для рисования
graph	private	Dictionary	Граф
	_	<string, string[] $>$	
friends	private	string[]	Список друзей
vertexRadius	private	float	Радиус отображаемой вершины
isAbleToForce	private	bool	Можно ли применить
			силовой алгоритм

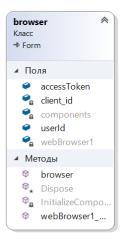
Таблица 10: Описание методов класса Form1

ObjToStr public string object obj Kactyet к строке					
ObjToStr public string object obj Kастует к строке Draw private void - Pисует граф на PictureBox Friends private String, string > Coздает список GetList private String, string - Cоздает список GetList private String - Cоздает список Друзей пользователя Возвращает общую UserInfo private string string id Bosspaщает общую Hopopmatipho o id Porrell private void - Cofestrue паменения ForeDirected private void - Cofestrue паменения ForeDirected private void - Cofestrue паменения ForeDirected private void - Cofestrue паменти Algorithm pasmemenus графа на Плоскости pasmemenus прафа на Плоскости pasmemenus прафа на Плоскости размешения клавиши мыши pictureBox private void - Cofestrue пажатия Kлавиши мыши клавиши мыши pictureBox private void - Cofestrue пажатия Kлавиши мыши pictureBox private void - Cofestrue пажатия MouseUp This private void - Ofepafortus события MouseDouble Click private void - Ofepafortus события Двойного нажатия мыши LPA тheoremCos private void - aлиоритм кластеризации LPA LPA возвращает половину Create private void - aлиоритм кластеризации Create private Edge - Cosjaer массив Dépafortus coferrus закрытия второй формы Create private Edge - Cosjaer массив Defection postate PointF int Huller координаты Regular PointF PointF Huller координаты LettreBox private PointF Int Huller координаты LettreBox private PointF Int Huller координаты LettreBox private PointF Int Huller paccroяние VerticesOf private double PointF Huller paccroяние	Имя	Модификатор	Тип	Аргументы	Назначение
Draw		·			
Friends GetMutual private Dictionary		_		object obj	_
SetMutual String, string SetList Private String String SetList Private String SetList Private String SetList Private String SetList Private String SetLing id Bosspaniaer общую информацию о id Pormi Private Void SizeChanged Pormi Private Pointf Pointf Pointf Private P		-		-	·
GetList FriendsByID string - Cоздает список друзей пользователя String id Boзвращает общую информацию о id Form1 private void - Cобытие изменения размера формы ForceDirected Algorithm Pointer void - Cобытие изменения размера формы Силовой алгоритм Pointer Point		private	· ·	<u> </u>	
FriendsByID UserInfo private string string id Bозвращает общую информацию о id Form1 private void - Событие изменения размера формы EroceDirected private void - Силовой алгоритм размещения графа на плоскости плоскоти плоскости плоскос					
UserInfo private string[] string id Возвращает общую информацию о id Form1_ SizeChanged private void - Событие изменения размера формы Силовой алгоритм размещения графа на плоскости ForeeDirected Algorithm private void - Силовой алгоритм размещения графа на плоскости pictureBox_ pictureBox_ private private void - Событие перемещения мыши MouseDown pictureBox_ MouseUp private void - Событие нажатия клавиши мыши MouseUp private void - Событие отжатия мыши MouseWheel pictureBox1_ MouseDouble Click private void - Обработчик события двойного нажатия мыши двойного нажатия мыши Clustering private void - алгоритм кластеризации LPA TheoremCos private int double R, double R, double a возвращает половину стороны многоугольника Method private void - Создает массив ребер графа Create private void - Возвращает граф в пачальное состояние Edges Array private		private	string[]		, ,
Form1_ SizeChanged	_				1 - 0
Form1_ SizeChanged	UserInfo	private	$\overline{\text{string}}$	string id	
SizeChanged размера формы ForceDirected Algorithm private void - Силовой алгоритм размещения графа на плоскости pictureBox_ MouseMove private void - Событие перемещения мыши pictureBox_ MouseDown private void - Событие пажатия клавиши мыши pictureBox_ MouseUp private void - Событие отжатия клавиши мыши This_ MouseWheel private void - Обработчик события прокрутки колесика мыши pictureBox1_ MouseDouble Click private void - Обработчик события двойного нажатия мыши Clustering private void - алгоритм кластеризации LPA TheoremCos private int double R, double R, double a стороны многоугольника Method private void - Обработчик события закрытия второй формы Create private Edge - Создает массив ребер графа ToStartPos private void - Возвращает граф в начальное состояние VerticesOf private PointF int n Ищет координаты центров кластеров Distanse private double PointF a, Ищет расстояние				<u> </u>	
ForceDirected Algorithmprivatevoid-Силовой алгоритм размещения графа на плоскостиpictureBox_ MouseMoveprivatevoid-Событие перемещения 		private	void		
ForceDirected Algorithmprivatevoid-Силовой алгоритм размещения графа на плоскостиpictureBox_ MouseMoveprivatevoid-Событие перемещения мышиpictureBox_ MouseDownprivatevoid-Событие пажатия клавиши мышиpictureBox_ MouseUpprivatevoid-Событие отжатия клавиши мышиThis_ MouseWheelprivatevoid-Обработчик события прокрутки колесика мышиpictureBox1_ MouseDouble Clickprivatevoid-Обработчик события двойного нажатия мышиClusteringprivatevoid-алгоритм кластеризации LPATheoremCosprivateintdouble R, double aвозвращает половину стороны многоугольникаMethodprivatevoid-Обработчик события закрытия второй формыCreateprivateEdge -Создает массив ребер графаToStartPosprivatevoid-Возвращает граф в начальное состояниеVerticesOfprivatePointF int nИщет росстояниеDistanseprivatedoublePointFa,Ищет расстояние				!	
Algorithmразмещения графа на плоскостиpictureBox_ MouseMoveprivatevoid- Событие перемещения мышиpictureBox_ MouseDownprivatevoid- Событие нажатия клавиши мышиpictureBox_ MouseUpprivatevoid- Событие отжатия клавиши мышиThis_ MouseWheelprivatevoid- Обработчик события прокрутки колесика мышиpictureBox1_ MouseDouble Clickprivatevoid- Обработчик события двойного нажатия мышиClickClusteringprivatevoid- алгоритм кластеризации LPATheoremCosprivateintdouble R, double aвозвращает половину стороны многоугольникаMethodprivatevoidОбработчик события закрытия второй формыCreateprivateEdge[]- Создает массив ребер графаToStartPosprivatevoid- Возвращает граф в начальное состояниеVerticesOfprivatePointF[]int nИщет координаты центров кластеровDistanseprivatedoublePointF a,Ищет расстояние	ForceDirected	private	void	_	
рісtureBox_ MouseMove void - Событие перемещения мыши рісtureBox_ MouseDown private void - Событие перемещения мыши рісtureBox_ MouseDown private void - Событие нажатия клавиши мыши рісtureBox_ MouseUp roivate void - Событие отжатия клавиши мыши This_ MouseWheel private void - Обработчик события прокрутки колесика мыши рісtureBox1_ private void - Обработчик события двойного нажатия мыши Click Clustering private void - алгоритм кластеризации LPA TheoremCos private int double R, возвращает половину стороны многоугольника Method private void Обработчик события закрытия второй формы Стеаte private void Обработчик события закрытия второй формы Стеаte private Edge - Создает массив ребер графа ToStartPos private void - Возвращает граф в начальное состояние VerticesOf private PointF int n Ищет координаты пентров кластеров Distanse private double PointFa, Ищет расстояние	Algorithm		l	1	
MouseMove мыши pictureBox_ MouseDown private void - Событие нажатия клавиши мыши pictureBox_ MouseUp private void - Событие отжатия клавиши мыши This_ MouseUp private void - Обработчик события прокрутки колесика мыши MouseWheel private void - Обработчик события двойного нажатия мыши MouseDouble Click void - алгоритм кластеризации LPA TheoremCos private int double R, double R, double a возвращает половину стороны многоугольника Method private void Обработчик события закрытия второй формы Create private void Обработчик события закрытия второй формы ToStartPos private Edge - Создает массив ребер графа ToStartPos private void - Возвращает граф в начальное состояние VerticesOf private PointF int n Ищет координаты центров кластеров Distanse private double PointF a, Ищет расстояние			l	1	плоскости
MouseMove pictureBox_ MouseDownprivatevoid-Событие нажатия клавиши мышиpictureBox_ MouseUpprivatevoid-Событие отжатия клавиши мышиThis_ MouseWheelprivatevoid-Обработчик события прокрутки колесика мышиpictureBox1_ MouseDouble Clickprivatevoid-Обработчик события двойного нажатия мышиClusteringprivatevoid-алгоритм кластеризации LPATheoremCosprivateintdouble R, double aвозвращает половину стороны многоугольникаMethodprivatevoidОбработчик события закрытия второй формыCreateprivatevoidОбработчик события закрытия второй формыCreateprivateEdge -Создает массив ребер графаToStartPosprivatevoid-Возвращает граф в начальное состояниеVerticesOfprivatePointF int nИщет координаты центров кластеровDistanseprivatedoublePointF a,Ищет расстояние	pictureBox_	private	void	_	Событие перемещения
MouseDownклавиши мышиpictureBox_ MouseUpprivatevoid-Событие отжатия клавиши мышиThis_ MouseWheelprivatevoid-Обработчик события прокрутки колесика мышиpictureBox1_ MouseDouble Clickprivatevoid-Обработчик события двойного нажатия мышиClusteringprivatevoid-алгоритм кластеризации LPATheoremCosprivateintdouble R, double R, double aвозвращает половину стороны многоугольникаMethodprivatevoidОбработчик события закрытия второй формы закрытия второй формыCreateprivateEdge[]-Создает массив ребер графаEdgesArraypeбер графав начальное состояниеVerticesOfprivatePointF[]int nИщет координаты центров кластеровDistanseprivatedoublePointF a,Ищет расстояние			!	1	мыши
MouseDown pictureBox_ MouseUpprivatevoid- Событие отжатия клавиши мышиThis_ MouseWheelprivatevoid- Обработчик события прокрутки колесика мышиpictureBox1_ MouseDouble Clickprivatevoid- Обработчик события двойного нажатия мышиClusteringprivatevoid- Апгоритм кластеризации LPATheoremCosprivateintdouble R, double aвозвращает половину стороны многоугольникаMethodprivatevoidОбработчик события закрытия второй формыCreate EdgesArrayprivateEdge[]-Создает массив ребер графаToStartPosprivatevoid-Возвращает граф в начальное состояниеVerticesOf RegularprivatePointF[]int nИщет координаты центров кластеровDistanseprivatedoublePointF a,Ищет расстояние	pictureBox_	private	void	_	Событие нажатия
MouseUpклавиши мышиThis_ MouseWheelprivatevoid-Обработчик события прокрутки колесика мышиpictureBox1_ MouseDouble Clickprivatevoid-Обработчик события двойного нажатия мышиClusteringprivatevoid-алгоритм кластеризации LPATheoremCosprivateintdouble R, double aвозвращает половину стороны многоугольникаMethodprivatevoidОбработчик события закрытия второй формыCreateprivateEdge[]-Создает массив ребер графаToStartPosprivatevoid-Возвращает граф в начальное состояниеVerticesOfprivatePointF[]int nИщет координаты центров кластеровDistanseprivatedoublePointF a,Ищет расстояние			l	1	клавиши мыши
MouseUpклавиши мышиThis_ MouseWheelprivatevoid-Обработчик события прокрутки колесика мыши прокрутки колесика мышиpictureBox1_ MouseDouble Clickprivatevoid-Обработчик события двойного нажатия мыши сыраClusteringprivatevoid-алгоритм кластеризации LPATheoremCosprivateintdouble R, double R, double aвозвращает половину стороны многоугольникаMethodprivatevoidОбработчик события закрытия второй формыCreateprivateEdge[]-Создает массив ребер графаToStartPosprivatevoid-Возвращает граф в начальное состояниеVerticesOfprivatePointF[]int nИщет координаты центров кластеровDistanseprivatedoublePointF a,Ищет расстояние	pictureBox_	private	void	_	Событие отжатия
This_ MouseWheelprivatevoid-Обработчик события прокрутки колесика мышиpictureBox1_ MouseDouble Clickprivatevoid-Обработчик события двойного нажатия мышиClusteringprivatevoid-алгоритм кластеризации LPATheoremCosprivateintdouble R, double aвозвращает половину стороны многоугольникаMethodprivatevoidОбработчик события закрытия второй формыCreateprivateEdge[]-Создает массив ребер графаToStartPosprivatevoid-Возвращает граф в начальное состояниеVerticesOfprivatePointF[]int nИщет координаты центров кластеровDistanseprivatedoublePointF a,Ищет расстояние		_	l	1	клавиши мыши
MouseWheelprivatevoid-Обработчик события двойного нажатия мышиDick-алгоритм кластеризации LPAClusteringprivatevoid-алгоритм кластеризации LPATheoremCosprivateintdouble R, double aвозвращает половину стороны многоугольникаMethodprivatevoidОбработчик события закрытия второй формыCreateprivateEdge-Создает массив ребер графаToStartPosprivatevoid-Возвращает граф в начальное состояниеVerticesOfprivatePointFint nИщет координаты центров кластеровDistanseprivatedoublePointF a,Ищет расстояние	This_	private	void	_	Обработчик события
pictureBox1_ MouseDouble Clickprivatevoid-Обработчик события двойного нажатия мышиClusteringprivatevoid-алгоритм кластеризации LPATheoremCosprivateintdouble R, double aвозвращает половину стороны многоугольникаMethodprivatevoidОбработчик события закрытия второй формыCreateprivateEdge[]-Создает массив ребер графаToStartPosprivatevoid-Возвращает граф в начальное состояниеVerticesOfprivatePointF[]int nИщет координаты центров кластеровDistanseprivatedoublePointF a,Ищет расстояние		_	l	1	_
MouseDouble Clickдвойного нажатия мышиClusteringprivatevoid-алгоритм кластеризации LPATheoremCosprivateintdouble R, double aвозвращает половину стороны многоугольникаMethodprivatevoidОбработчик события закрытия второй формыCreateprivateEdge[]-Создает массив ребер графаToStartPosprivatevoid-Возвращает граф в начальное состояниеVerticesOfprivatePointF[]int nИщет координаты центров кластеровDistanseprivatedoublePointF a,Ищет расстояние	pictureBox1_	private	void	_	
Clusteringprivatevoid-алгоритм кластеризации LPATheoremCosprivateintdouble R, double R, double aвозвращает половину стороны многоугольникаMethodprivatevoidОбработчик события закрытия второй формыCreateprivateEdge[]-Создает массив ребер графаEdgesArraypeбер графав начальное состояниеVerticesOfprivatePointF[]int nИщет координаты центров кластеровDistanseprivatedoublePointF a,Ищет расстояние	1	_	!	1	двойного нажатия мыши
LPATheoremCosprivateintdouble R, double R, double aвозвращает половину стороны многоугольникаMethodprivatevoidОбработчик события закрытия второй формыCreateprivateEdge[]-Создает массив ребер графаEdgesArraypeбер графав начальное состояниеToStartPosprivatevoid-Возвращает граф в начальное состояниеVerticesOfprivatePointF[]int nИщет координаты центров кластеровDistanseprivatedoublePointF a,Ищет расстояние	Click		l	1	
TheoremCosprivateintdouble R, double R, double R, double aвозвращает половину стороны многоугольникаMethodprivatevoidОбработчик события закрытия второй формыCreateprivateEdge[]-Создает массив ребер графаEdgesArraypeбер графа-Возвращает граф в начальное состояниеVerticesOfprivatePointF[]int nИщет координаты центров кластеровDistanseprivatedoublePointF a,Ищет расстояние	Clustering	private	void	_	алгоритм кластеризации
MethodprivatevoidОбработчик события закрытия второй формыCreateprivateEdge[]-Создает массив ребер графаEdgesArrayToStartPosprivatevoid-Возвращает граф в начальное состояниеVerticesOfprivatePointF[]int nИщет координаты центров кластеровDistanseprivatedoublePointF a,Ищет расстояние		_	l	1	
MethodprivatevoidСтороны многоугольникаCreateprivateEdge[]-Создает массивEdgesArraypeбер графаToStartPosprivatevoid-Возвращает графVerticesOfprivatePointF[]int nИщет координатыRegularprivatedoublePointF a,Ищет расстояние	TheoremCos	private	int	double R,	возвращает половину
CreateprivateEdge[]-Создает массивEdgesArraypeбер графаToStartPosprivatevoid-Возвращает графVerticesOfprivatePointF[]int nИщет координатыRegularпентров кластеровDistanseprivatedoublePointF a,Ищет расстояние			!	1 ' 1	
CreateprivateEdge[]-Создает массивEdgesArraypeбер графаToStartPosprivatevoid-Возвращает графVerticesOfprivatePointF[]int nИщет координатыRegularпентров кластеровDistanseprivatedoublePointF a,Ищет расстояние	Method	private	void		Обработчик события
Create EdgesArrayprivateEdge[]-Создает массив ребер графаToStartPosprivatevoid-Возвращает граф в начальное состояниеVerticesOf RegularprivatePointF[]int nИщет координаты центров кластеровDistanseprivatedoublePointF a,Ищет расстояние		-	!	1	закрытия второй формы
EdgesArraypeбер графаToStartPosprivatevoid-Возвращает граф в начальное состояниеVerticesOfprivatePointF[]int nИщет координаты центров кластеровDistanseprivatedoublePointF a,Ищет расстояние	Create	private	Edge[]	_	
ToStartPos private void - Возвращает граф в начальное состояние VerticesOf private PointF[] int п Ищет координаты центров кластеров Distanse private double PointF a, Ищет расстояние	EdgesArray	-	- <u></u>	1	ребер графа
VerticesOf private PointF[] int n Ищет координаты центров кластеров Regular Distanse private double PointF a, Ищет расстояние		private	void	_	Возвращает граф
Regular центров кластеров Distanse private double PointF a, Ищет расстояние		-	!	1	в начальное состояние
Regular центров кластеров Distanse private double PointF a, Ищет расстояние	VerticesOf	private	PointF[]	int n	Ищет координаты
Distanse private double PointF a, Ищет расстояние	Regular	-		1	
		private	double	PointF a,	
· · · · · · · · · · · · · · · · · · ·		_	l	1 ' 1	между точками

6 Диаграмма классов







7 Лист регистрации изменений

Лист регистрации изменений									
Изм.	Номера листов				Всего листов в доку- менте	№ доку- мента	Входящий № сопро- водит. докум. и дата	Под- пись	Дата
	из- ме- нен- ных	заме- нен- ных	НО- ВЫХ	аннул- лиро- ван- ных					