

SECTION 5. Innovative technologies in science.

Vladimir Nikolayevich Kestelman

Doctor of technical Sciences, Professor, Specialty Scientific Consultant, Russian Technology Initiative, LTD, President of KVN International, Inc., Member of Mid-Atlantic-Russia Business Council and of International Visitors Council of Philadelphia, King of Prussia, Pennsylvania, USA

Alexandr Nikolayevich Shevtsov

candidate of technical Sciences, President of International Academy International Academy of Theoretical & Applied Sciences (USA, Sweden, Kazakhstan)
Shev_AlexXXXX@mail.ru

Altynay Kayratovna Akhmetkaliyeva

student
Taraz State University named after M.Kh. Dulati, Kazakhstan

THREE-DIMENSIONAL TOPOLOGICAL CODING WITH A PRIVATE KEY ON THE MOBIUS STRIP

***Abstract:** In this paper we consider one approach encoding information on topological surface - when to encode information easy enough, but to return the original data on the basis of three-dimensional the surface is almost impossible. Meanwhile the 3D surface is nothing like a Mobius strip broken through triangulation. At present it is necessary to develop and explore topology, and aim to link its achievements with the real world in practical tasks.*

***Key words:** the Mobius strip, coding, topology, 3D, modeling.*

ТОПОЛОГИЧЕСКОЕ ТРЕХМЕРНОЕ КОДИРОВАНИЕ С ЗАКРЫТЫМ КЛЮЧЕМ НА ЛЕНТЕ МЕБИУСА

***Аннотация:** В данном исследовании предлагается один из подходов кодирования информации по топологической поверхности, – когда закодировать информацию достаточно просто, а вот вернуть исходные данные на основе трехмерной поверхности – почти невозможно. Между тем сама 3D поверхность – это не что иное, как лента Мебиуса разбитая с помощью триангуляции. В настоящее время необходимо развивать и исследовать топологию, стремясь связать ее достижения с реальным миром в практических задачах.*

***Ключевые слова:** лента Мебиуса, кодирование, топология, 3D, моделирование.*

Введение

Активное развитие топологии и изучение топологических принципов различных объектов все чаще находит применение в повседневной жизни. Развитая компьютерная инфраструктура учебных заведений – только способствует внедрению и использованию накопленного математического опыта последних веков.

Лента Мебиуса была открыта в 1858 году, но несмотря на это, она может принести еще очень много новых и совершенно непредсказуемых открытий.

В данном исследовании рассматривается один из подходов кодирования информации – когда закодировать информацию достаточно просто, а вот вернуть исходные данные на основе трехмерной поверхности – почти невозможно. Между тем

сама 3D поверхность – это не что иное, как лента Мебиуса разбитая с помощью триангуляции.

Цель работы: Разработка методов кодирования на основе трехмерных топологических объектов.

Актуальность: В настоящее время необходимо не просто исследовать топологию, как мало изученный раздел математики, а стремиться связать ее достижения с реальным миром и внедрить в производство. Подобная модель разрабатывается – впервые.

Внедрение: Полученная модель позволяет быстро и удобно кодировать текст на русском и английском языках, при необходимости можно дополнить и другими. Модель реализованна в готовый интерактивный программный продукт. Разработанные алгоритмы могут использоваться в различных сферах кодирования информации.

ЛЕНТА МЁБИУСА

Лента Мёбиуса - или как ее еще называют - Лист Мёбиуса (петля Мёбиуса) — топологический объект, простейшая неориентируемая поверхность с краем, односторонняя при вложении в обычное трёхмерное Евклидово пространство \mathbb{R}^3 . Попастъ из одной точки этой поверхности в любую другую можно, не пересекая края.[1]



Рисунок 1 – Лента Мебиуса.

Лента Мёбиуса была открыта независимо немецкими математиками Августом Фердинандом Мёбиусом и Иоганном Бенедиктом Листингом в 1858 году. Модель ленты Мёбиуса может легко быть сделана: для этого надо взять достаточно вытянутую бумажную полоску и соединить концы полоски, предварительно перевернув один из них. В Евклидовом пространстве существуют два типа полос Мёбиуса в зависимости от направления закручивания: правые и левые (топологически они, все же, неразличимы) [1]

Если разрезать ленту Мебиуса вдоль по линии, равноудалённой от краёв, вместо двух лент Мёбиуса получится одна длинная двухсторонняя (вдвое больше закрученная, чем лента Мёбиуса) лента, которую называют «Афганская лента». Если теперь эту ленту разрезать вдоль посередине, получаютcя две ленты, намотанные друг на друга.[1,4]

Если разрезать ленту Мёбиуса, отступая от края приблизительно на треть её ширины, то получаются две ленты, одна — более короткая лента Мёбиуса, другая — длинная лента с двумя полуоборотами (афганская лента).[1-2]

Другие комбинации лент могут быть получены из лент с двумя или более полуоборотами в них. Например, если разрезать ленту с тремя полуоборотами, то получится лента, завитая в узел трилистника. Разрез ленты с дополнительными оборотами даёт неожиданные фигуры, названные парадоксными кольцами.[1]

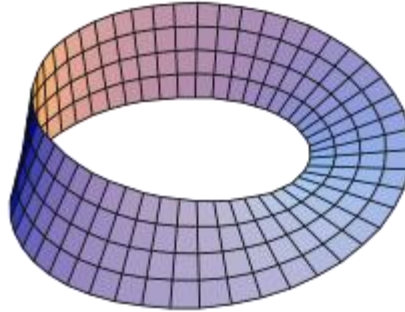


Рисунок 2 - Модель ленты.

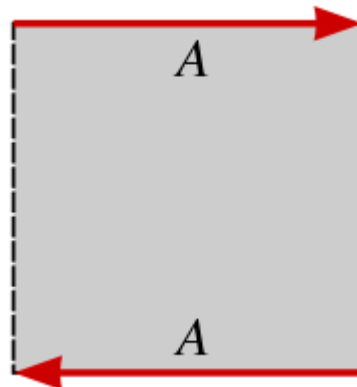


Рисунок 3 - Параметрическое описание листа Мёбиуса.

Чтобы превратить квадрат в лист Мёбиуса, соедините края, таким образом, чтобы направления стрелок совпали. Одним из способов представления листа Мёбиуса как подмножества \mathbb{R}^3 является параметризация:[1-7]

$$\begin{aligned} x(u, v) &= \left(1 + \frac{v}{2} \cos \frac{u}{2}\right) \cos u, \\ y(u, v) &= \left(1 + \frac{v}{2} \cos \frac{u}{2}\right) \sin u, \\ z(u, v) &= \frac{v}{2} \sin \frac{u}{2}, \end{aligned} \quad (1)$$

где $0 \leq u < 2\pi$ и $-1 \leq v \leq 1$. Эти формулы задают ленту Мёбиуса ширины 1, чей центральный круг имеет радиус 1, лежит в плоскости $x - y$ с центром в $(0, 0, 0)$. Параметр u пробегает вдоль ленты, в то время как v задает расстояние от края.[1]

В цилиндрических координатах (r, θ, z) неограниченная версия листа Мёбиуса может быть представлена уравнением:

$$\log r \sin(\theta/2) = z \cos(\theta/2),$$

где функция логарифма имеет произвольное основание.[1]

Топологически лист Мёбиуса может быть определен как факторпространство квадрата $[0, 1] \times [0, 1]$ по отношению эквивалентности $(x, 0) \sim (1 - x, 1)$ для $0 \leq x \leq 1$. Лист Мёбиуса — неориентируемая поверхность с краем.

Существуют технические применения ленты Мёбиуса. Полоса ленточного конвейера выполняется в виде ленты Мёбиуса, что позволяет ему работать дольше, потому что вся поверхность ленты изнашивается равномерно. Также в системах записи на непрерывную плёнку применяются ленты Мёбиуса (чтобы удвоить время записи). Во многих матричных принтерах красящая лента также имеет вид листа Мёбиуса для увеличения её ресурса.[1-5]

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ТОПОЛОГИЧЕСКОГО 3D КОДИРОВАНИЯ

За основу кодирования будем использовать следующий алфавит:

```
const
a='ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789/*+
.,~!@#$%^&()_абвгдеёжзийклмнопрстуфхцчшщъыьэю
Яабвгдеёжзийклмнопрстуфхцчшщъыьэюя';
```

Как бы то ни было, но в 1858 году Лейпцигский профессор Август Фердинанд Мёбиус, ученик знаменитого К.Ф.Гаусса, астроном и геометр, послал в Парижскую академию наук работу, включающую сведения об этом листе. Семь лет он дождался рассмотрения своей работы, и не дождавшись, опубликовал её результатов в 1858 году.
У листа Мёбиуса всего одна сторона, и это поразило немецких профессоров, и потому что каждая поверхность имеет две стороны.
Лист Мёбиуса
Рассказывают, что открыть свой "лист Мёбиуса" помогла служанка шившая

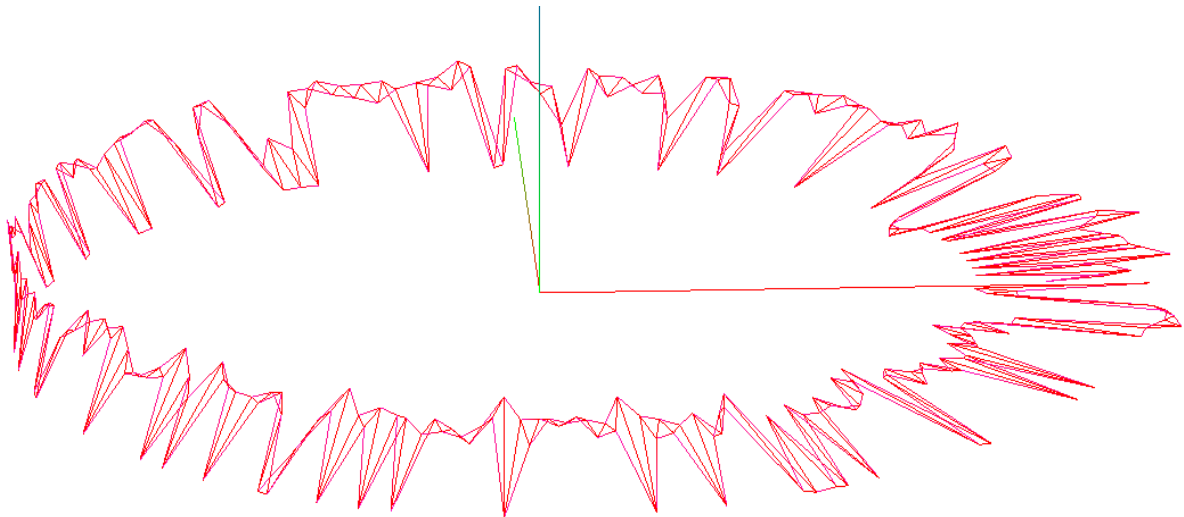
Кодировать

Рисунок 4 – Исходный текст.

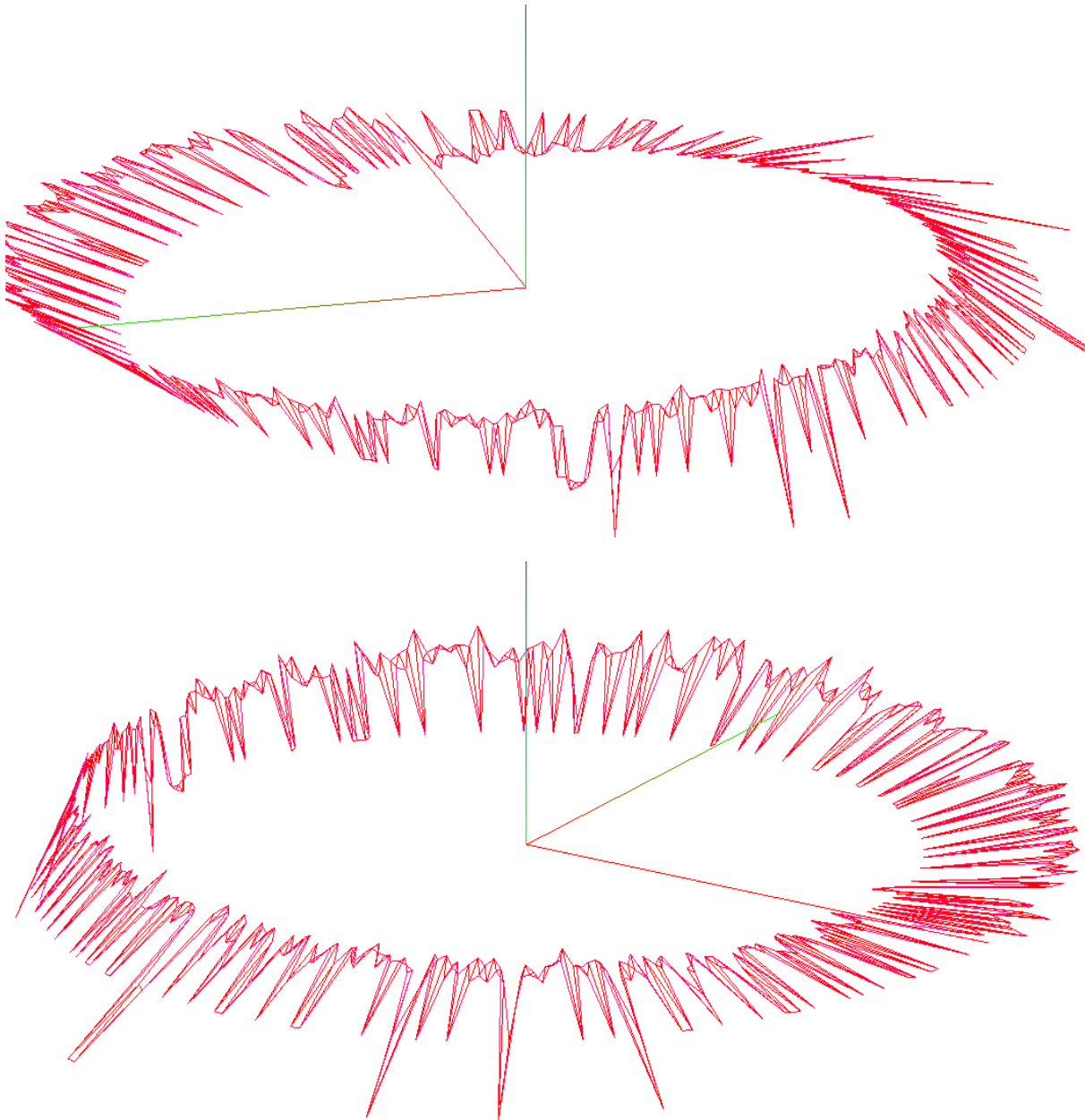
Текст равномерно распределим по окружности в плоскости Оху. Каждый символ алфавита будем смещать по нормали к направлению движения на заданный порядок – равный индексу буквы в алфавите. Координаты точки в трехмерном пространстве будем определять по формулам (1). Самое главное – то что криптостойкость обеспечивается – сложностью нахождения обратной функции к системе (1) и дает не только одно значение – а сразу несколько (Приложение 1). Причем неизвестен угол начального поворота ленты и ее радиус . Что значительно усложняет обратную задачу.[8-24]

```
Kod.txt — Блокнот
Файл  Правка  Формат  Вид  Справка
1,1448734998703
0,0167584978044033
0,00106115953531116
-----
1,11985337734222
0,0178827047348022
0,000958035874646157
-----
0,834881246089935
0,0144432615488768
-0,00142707163468003
-----
1,07980918884277
0,0201177392154932
0,000745136174373329
-----
1,21474730968475
0,0242487173527479
0,00214558839797974
-----
1,13473510742188
0,0241620615124702
0,00143704190850258
-----
1,14969623088837
0,0260111764073372
0,00169650372117758
-----
0,844768762588501
0,0202370546758175
-----
Стр 1, столб 1
```

Рисунок 5 – Зашифрованные данные.



**Рисунок 6 – Трехмерный аналог закодированного текста,
в форме ленты Мебиуса (314 символов)**



**Рисунок 7 – Трехмерный аналог закодированного текста,
Лента Мебиуса (764 символов)**

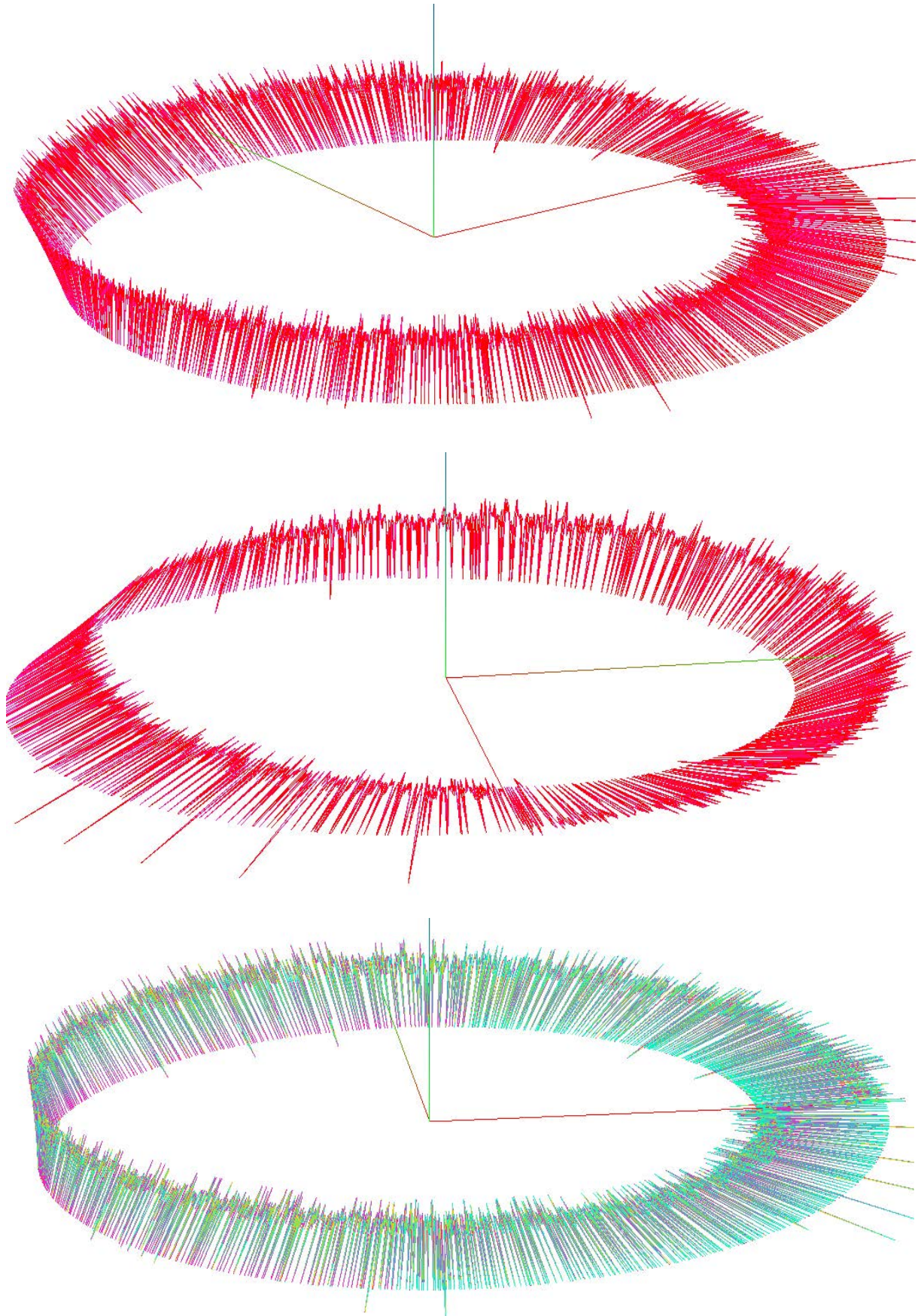


Рисунок 8 – Трехмерный аналог закодированного текста, Лента Мебиуса (4722 СИМВОЛОВ)

Выводы

Разработанная математическая модель (рис.6-8) связала топологический объект – Ленту Мебиуса, с алгоритмом алфавитного кодирования. Закрытый ключ в данном случае представляют собой – коэффициенты системы (1), центр ленты Мебиуса и ее ширина. Алгоритм может считаться односторонним в связи со сложностью обратного расчета системы (1) (приложение 1) и отсутствия единственного решения (наличие периода), а также неопределенность начального угла смещения.

В результате проведенного исследования получены следующие результаты:

- Разработана математическая модель кодирования данных,
- Реализовано трехмерное преобразование по поверхности ленты Мебиуса,
- Созданы алгоритмы для практической реализации математической модели,
- Сделана компьютерная реализация модели кодирования, и декодирования
- Разработаны алгоритмы интерактивного 3D отображения ленты Мебиуса в программе,
- Модель реализованна в готовый интерактивный программный продукт.
- Разработанные алгоритмы могут использоваться в различных сферах кодирования информации.

Разработанная модель имеет продолжение в большем изгибе в тороидальную поверхность, а также возможно спиралевидное смещение по оси Oz, при этом будет получен цилиндр трехмерных координат.

Особенно интересная задача может быть построена в случае выбора минимизирующих точек на перпендикулярной плоскости. Причем сам выбор будет делаться таким образом – чтобы выделить точки пересечения плоскостей. В этом случае – любой текст может быть сжат до одной единственной точки в пространстве.

References:

1. Лента Мёбиуса. [Электронный ресурс] URL: http://ru.wikipedia.org/wiki/%CB%E5%ED%F2%E0_%CC%B8%E1%E8%F3%F1%E0 (дата доступа 10.04.2014)
2. Лист Мебиуса. [Электронный ресурс] URL: <http://nsportal.ru/ap/nauchno-tehnicheskoe-tvorchestvo/list-mebiusa> (дата доступа 10.04.2014)
3. Лист Мёбиуса и его свойства. [Электронный ресурс] URL: <http://ru.convdocs.org/docs/index-41104.html> (дата доступа 10.04.2014)
4. Лист Мёбиуса. [Электронный ресурс] URL: http://www.coolreferat.com/%D0%9B%D0%B8%D1%81%D1%82_%D0%9C%D0%B5%D0%B1%D0%B8%D1%83%D1%81%D0%B0 (дата доступа 10.04.2014)
5. Решение уравнений и неравенств. Корни многочленов. [Электронный ресурс] URL: http://www.exponenta.ru/soft/Maple/manson/poweredition/chapter6/6_2_3.asp (дата доступа 10.04.2014)
6. 3D поверхности. [Электронный ресурс] URL: <http://www.programmersforum.ru/showthread.php?t=193671> (дата доступа 10.04.2014)
7. Учимся работать с DelphiX. [Электронный ресурс] URL: http://www.delphigfx.narod.ru/2d/013/2d_13.htm (дата доступа 10.04.2014)

8. Построение 3d поверхности через OpenGL, нужна помощь, информация, примеры. [Электронный ресурс] URL: <http://forum.codeby.net/topic24923.html> (дата доступа 16.04.2014)
9. The Moebius Strip. [Электронный ресурс] URL: http://mathforum.org/sum95/math_and/moebius/moebius.html (дата доступа 16.04.2014)
10. Рисуем простые объекты. [Электронный ресурс] URL: http://www.helloworld.ru/texts/comp/games/opengl/opengl2/opengl_03.html (дата доступа 16.04.2014)
11. Отображение полигонов. [Электронный ресурс] URL: <http://pmg.org.ru/nehe/nehe02.htm> (дата доступа 16.04.2014)
12. Координаты и всё, что с ними связано. [Электронный ресурс] URL: http://delphiworld.narod.ru/base/ogl_beginners2.html (дата доступа 16.04.2014)
13. delphi/OpenGL нужна помощь... [Электронный ресурс] URL: <http://www.programmersforum.ru/showthread.php?t=74641> (дата доступа 16.04.2014)
14. glVertexAttribPointerARB [Delphi, Windows] [Электронный ресурс] URL: <http://www.delphimaster.ru/cgi-bin/forum.pl?id=1213600883&n=9> (дата доступа 16.04.2014)
15. Простые примитивы [Электронный ресурс] URL: <http://thedelphi.ru/opengl/oles2.php> (дата доступа 16.04.2014)
16. Основы Delphi. [Электронный ресурс] URL: <http://www.delphibasics.ru/Single.php> (дата доступа 16.04.2014)
17. How to fill polygon with different color than boundary? [Электронный ресурс] URL: <http://stackoverflow.com/questions/5146216/how-to-fill-polygon-with-different-color-than-boundary> (дата доступа 16.04.2014)
18. Буфер. [Электронный ресурс] URL: <http://www.helloworld.ru/texts/comp/games/opengl/opengl/2.htm> (дата доступа 16.04.2014)
19. Освещение и все что с ним связано. [Электронный ресурс] URL: http://citforum.ru/programming/opengl/opengl_06.shtml (дата доступа 16.04.2014)
20. Источники света. [Электронный ресурс] URL: http://compgraphics.info/OpenGL/lighting/light_sources.php (дата доступа 16.04.2014)
21. glViewport и его параметры. [Электронный ресурс] URL: <http://www.programmersforum.ru/showthread.php?t=78650> (дата доступа 16.04.2014)
22. ViewPort. [Электронный ресурс] URL: <http://www.firststeps.ru/mfc/opengl/r.php?17> (дата доступа 16.04.2014)
23. ColorToRGB Routine. [Электронный ресурс] URL: http://delphi.wikia.com/wiki/ColorToRGB_Routine (дата доступа 16.04.2014)
24. Преобразовать RGB в TColor и наоборот. [Электронный ресурс] URL: http://www.delphisources.ru/pages/faq/base/rgb_tcolor.html (дата доступа 16.04.2014)

Приложение 1

Решение системы (1) в Maple и постановка обратной задачи

> restart;

```
R1:=x=(1+v/2*cos(u/2))*cos(u);
R2:=y=(1+v/2*cos(u/2))*sin(u);
R3:=z=v/2*sin(u/2);
```

$$R1 := x = \left(1 + \frac{1}{2} v \cos\left(\frac{1}{2} u\right) \right) \cos(u)$$

$$R2 := y = \left(1 + \frac{1}{2} v \cos\left(\frac{1}{2} u\right) \right) \sin(u)$$

$$R3 := z = \frac{1}{2} v \sin\left(\frac{1}{2} u\right)$$

> v:=solve(R3,v);

$$v := 2 \frac{z}{\sin\left(\frac{1}{2} u\right)}$$

> R1;R2;

$$x = \left(1 + \frac{z \cos\left(\frac{1}{2} u\right)}{\sin\left(\frac{1}{2} u\right)} \right) \cos(u)$$

$$y = \left(1 + \frac{z \cos\left(\frac{1}{2} u\right)}{\sin\left(\frac{1}{2} u\right)} \right) \sin(u)$$

> R4:=solve(R1,u);

$$R4 := 2 \arctan\left(\frac{z \%1 (2 \%1^2 - 1)}{x - 2 \%1^2 + 1}, \%1\right)$$

%1 :=

$$\text{RootOf}((4z^2 + 4)_Z^6 + (-4x - 4z^2 - 8)_Z^4 + (5 + 6x + z^2 + x^2)_Z^2 - x^2 - 2x - 1)$$

> R5:=solve(R2,u);

$$R5 := 2 \arctan\left(\frac{1}{2} \frac{y - 2z \%1^2}{\%1}, \%1\right)$$

$$\%1 := \text{RootOf}((4z^2 + 4) _Z^4 + (-4yz - 4) _Z^2 + y^2)$$

> S:=allvalues(R4);

$$S := 2 \arctan\left(\frac{z \sqrt{\%9} \left(-12 \frac{\%2^{1/3} (z^2 + 1) \%7}{\%8^2} - 1\right)}{\%8 \left(x + 12 \frac{\%2^{1/3} (z^2 + 1) \%7}{\%8^2} + 1\right)}, \frac{\sqrt{\%9}}{\%8}\right)$$

$$2 \arctan\left(\frac{z \sqrt{\%9} \left(-12 \frac{\%2^{1/3} (z^2 + 1) \%7}{\%8^2} - 1\right)}{\%8 \left(x + 12 \frac{\%2^{1/3} (z^2 + 1) \%7}{\%8^2} + 1\right)}, -\frac{\sqrt{\%9}}{\%8}\right)$$

$$2 \arctan\left(2 \frac{z \sqrt{3} \sqrt{(z^2 + 1) \%2^{1/3} \%6} \left(24 \frac{(z^2 + 1) \%2^{1/3} \%6}{\%4^2} - 1\right)}{\%4 \left(x - 24 \frac{(z^2 + 1) \%2^{1/3} \%6}{\%4^2} + 1\right)}, 2 \frac{\sqrt{3} \sqrt{(z^2 + 1) \%2^{1/3} \%6}}{\%4}\right)$$

$$2 \arctan\left(-2 \frac{z \sqrt{3} \sqrt{(z^2 + 1) \%2^{1/3} \%6} \left(24 \frac{(z^2 + 1) \%2^{1/3} \%6}{\%4^2} - 1\right)}{\%4 \left(x - 24 \frac{(z^2 + 1) \%2^{1/3} \%6}{\%4^2} + 1\right)}, -2 \frac{\sqrt{3} \sqrt{(z^2 + 1) \%2^{1/3} \%6}}{\%4}\right)$$

$$2 \arctan\left(2 \frac{z \sqrt{\%5} \left(-24 \frac{\%2^{1/3} (z^2 + 1) \%3}{\%4^2} - 1\right)}{\%4 \left(x + 24 \frac{\%2^{1/3} (z^2 + 1) \%3}{\%4^2} + 1\right)}, 2 \frac{\sqrt{\%5}}{\%4}\right)$$

$$2 \arctan \left(-2 \frac{z \sqrt{\%5} \left(-24 \frac{\%2^{1/3} (z^2 + 1) \%3}{\%4^2} - 1 \right)}{\%4 \left(x + 24 \frac{\%2^{1/3} (z^2 + 1) \%3}{\%4^2} + 1 \right)} \right), -2 \frac{\sqrt{\%5}}{\%4}$$

$$\%1 = -z^4 + 14z^2x + 11z^2x^2 + 2z^2 - 1 + 2x + x^4 - 2x^3$$

$$\%2 = -3x - 12z^2x - 3z^2x^2 - x^3 + 3x^2 - 3z^2 + 3z^4 - z^6 - 9x^3z^2 + 15z^4x + 18z^4x^2 + 1 + 3\sqrt{3}z^3x\sqrt{\%1} + 3\sqrt{3}zx\sqrt{\%1}$$

$$\%3 = I\sqrt{3}\%2^{2/3} + 2I\sqrt{3}z^2 + \%2^{2/3} - 2z^2 + 1 - 10z^2x - 2x + z^4 - 3z^2x^2 + x^2 - 4\%2^{1/3}x - 4\%2^{1/3}z^2 - 8\%2^{1/3} + 2I\sqrt{3}x - I\sqrt{3}z^4 - I\sqrt{3} + 10I\sqrt{3}z^2x + 3I\sqrt{3}z^2x^2 - I\sqrt{3}x^2$$

$$\%4 = 12\%2^{1/3}z^2 + 12\%2^{1/3}$$

$$\%5 = -3\%2^{1/3}(z^2 + 1)\%3$$

$$\%6 = I\sqrt{3}\%2^{2/3} + 2I\sqrt{3}z^2 - \%2^{2/3} + 2z^2 - 1 + 10z^2x + 2x - z^4 + 3z^2x^2 - x^2 + 4\%2^{1/3}x + 4\%2^{1/3}z^2 + 8\%2^{1/3} + 2I\sqrt{3}x - I\sqrt{3}z^4 - I\sqrt{3} + 10I\sqrt{3}z^2x + 3I\sqrt{3}z^2x^2 - I\sqrt{3}x^2$$

$$\%7 = -1 + 10z^2x - \%2^{2/3} + 2z^2 + 3z^2x^2 - x^2 + 2x - z^4 - 4\%2^{1/3} - 2\%2^{1/3}x - 2\%2^{1/3}z^2$$

$$\%8 = 6\%2^{1/3} + 6\%2^{1/3}z^2$$

$$\%9 = -6\%2^{1/3}(z^2 + 1)\%7$$

> S2:=allvalues(R5);

$$S2 := 2 \arctan \left(\frac{1}{4} \frac{\left(y - 4 \frac{z(z^2 + 1)\%2}{2} \right) (2 + 2z^2) \sqrt{2}}{\sqrt{(z^2 + 1)\%2}}, \frac{\sqrt{2} \sqrt{(z^2 + 1)\%2}}{2 + 2z^2} \right)$$

$$2 \arctan \left(-\frac{1}{4} \frac{\left(y - 4 \frac{z(z^2 + 1)\%2}{2} \right) (2 + 2z^2) \sqrt{2}}{\sqrt{(z^2 + 1)\%2}}, -\frac{\sqrt{2} \sqrt{(z^2 + 1)\%2}}{2 + 2z^2} \right)$$

$$2 \arctan \left(\frac{1}{4} \frac{\left(y - 4 \frac{z(z^2 + 1)\%1}{2} \right) (2 + 2z^2) \sqrt{2}}{\sqrt{(z^2 + 1)\%1}}, \frac{\sqrt{2} \sqrt{(z^2 + 1)\%1}}{2 + 2z^2} \right)$$

$$2 \arctan \left(-\frac{1}{4} \frac{\left(y - 4 \frac{z(z^2 + 1)\%1}{2} \right) (2 + 2z^2) \sqrt{2}}{\sqrt{(z^2 + 1)\%1}}, -\frac{\sqrt{2} \sqrt{(z^2 + 1)\%1}}{2 + 2z^2} \right)$$

$$\%1 := yz + 1 - \sqrt{2yz + 1 - y^2}$$

$$\%2 := yz + 1 + \sqrt{2yz + 1 - y^2}$$

Приложение 2

Код программы в Delphi (кодирование)

```

unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, ExtCtrls, OpenGL, ComCtrls, StdCtrls, Spin, math, Vcl.ColorGrd;
type
  TVector = Array[0..2] of single; // тут храним X, Y, Z

type
  TForm1 = class(TForm)
    Timer1: TTimer;
    Timer2: TTimer;
    Panel1: TPanel;
    Button1: TButton;
    Memo1: TMemo;
    TrackBar1: TTrackBar;
    Button3: TButton;
    Memo2: TMemo;
    ColorGrid1: TColorGrid;
    CheckBox1: TCheckBox;
    procedure FormCreate(Sender: TObject);
    procedure FormResize(Sender: TObject);
    procedure Timer1Timer(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    procedure Timer2Timer(Sender: TObject);
    procedure FormDblClick(Sender: TObject);
    procedure Button1Click(Sender: TObject);
    procedure FormDestroy(Sender: TObject);
    procedure TrackBar1Change(Sender: TObject);
    procedure Button3Click(Sender: TObject);
    function Mebius(u,v:double):Tvector;
    procedure ColorGrid1Change(Sender: TObject);
  private
    ghRC:HGLRC;
    ghDC:HDC;
    procedure Draw;
  public
    { Public declarations }
  end;
var

```



```
Form1: TForm1;
d,r,rx,ry,dd,fi:double;
kr,n:integer;
var
  xyz:array[0..10000000] of TVector;
  col2:tcolor;
implementation

{$R *.dfm}
function bSetupPixelFormat(DC:HDC):boolean;
var
  pfd:PIXELFORMATDESCRIPTOR;
  ppfd:PPIXELFORMATDESCRIPTOR;
  pixelformat:integer;
begin
  ppfd := @pfd;

  ppfd.nSize := sizeof(PIXELFORMATDESCRIPTOR);
  ppfd.nVersion := 1;
  ppfd.dwFlags := PFD_DRAW_TO_WINDOW xor
    PFD_SUPPORT_OPENGL xor
    PFD_DOUBLEBUFFER;
  ppfd.dwLayerMask := PFD_MAIN_PLANE;
  ppfd.iPixelFormat := PFD_TYPE_RGBA;
  ppfd.cColorBits := 16;
  ppfd.cDepthBits := 16;
  ppfd.cAccumBits := 0;
  ppfd.cStencilBits := 0;

  pixelformat := ChoosePixelFormat(dc, ppfd);
  if pixelformat=0 then
    begin
      MessageBox(0, 'ChoosePixelFormat failed', 'Error', MB_OK);
      bSetupPixelFormat:=FALSE;
      exit;
    end;

  if SetPixelFormat(dc, pixelformat, ppfd)=false then
    begin
      MessageBox(0, 'SetPixelFormat failed', 'Error', MB_OK);
      bSetupPixelFormat:=FALSE;
      exit;
    end;

  bSetupPixelFormat:=TRUE;
end;

procedure TForm1.FormCreate(Sender: TObject);
var
  p: TGLArrayf4;
  d: TGLArrayf3;
```

```

I: Integer;
begin

kr:=3;
  dd:=1;
  ghDC := GetDC(Handle);
  if bSetupPixelFormat(ghDC)=false then    Close();
  ghRC := wglCreateContext(ghDC);
  wglMakeCurrent(ghDC, ghRC);

  glClearColor(4.0, 4.0, 4.0, 4.0);

  FormResize(Sender);

  glEnable(GL_COLOR_MATERIAL);
  glEnable(GL_DEPTH_TEST);

  glEnable(GL_LIGHTING);
  glEnable(GL_LIGHT0);
  p[0]:=3;
  p[1]:=3;
  p[2]:=3;
  p[3]:=0;
  d[0]:=0;
  d[1]:=0;
  d[2]:=-3;
  glLightfv(GL_LIGHT0,GL_POSITION,@p);
  glLightfv(GL_LIGHT0,GL_SPOT_DIRECTION,@d);

  glEnable(GL_LIGHT1);
  p[0]:=-3;
  p[1]:=-3;
  p[2]:=3;
  p[3]:=0;
  d[0]:=0;
  d[1]:=0;
  d[2]:=-3;
  glLightfv(GL_LIGHT1,GL_POSITION,@p);
  glLightfv(GL_LIGHT1,GL_SPOT_DIRECTION,@d);

Button3.Click;
  end;

procedure TForm1.FormDbClick(Sender: TObject);
begin
application.Terminate;
end;

procedure TForm1.FormResize(Sender: TObject);
begin
  glViewport( 100, 0, Width, Height );

```

```

glMatrixMode( GL_PROJECTION );
glLoadIdentity();
glOrtho(-kr/2,kr/2, -kr/2,kr/2, 0,30);

gluLookAt( 2,2,2, 0,0,0, 0,0,10);
glMatrixMode( GL_MODELVIEW );

end;

procedure TForm1.Button1Click(Sender: TObject);
begin
  Timer1.Enabled:=not(Timer1.Enabled);
  if not(Timer1.Enabled) then exit;

  Button3.Click;
  Draw;
end;

function TForm1.Mebius(u,v:double):Tvector;
begin
  xyz[n][0]:= (1+(v/2)*cos(u/2*pi/180))*cos(u*pi/180);
  xyz[n][1]:= (1+(v/2)*cos(u/2*pi/180))*sin(u*pi/180);
  xyz[n][2]:= (v/2)*sin(u/2*pi/180);
  form1.memo2.Lines.Add(floattostr(xyz[n][0]));
  form1.memo2.Lines.Add(floattostr(xyz[n][1]));
  form1.memo2.Lines.Add(floattostr(xyz[n][2]));
  form1.memo2.Lines.Add('_____');
end;

function ABCD(s:string):integer;
const
  a='ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789/*-+
.,~!@#$$%^&()_ =АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯабвгдеёжзийклм
нопрстуфхцчшщъыьэюя';
var i:integer;
begin
  for i:=1 to length(a) do
  if s=a[i] then
  begin
    ABCD:=i;
    exit;
  end;
end;

procedure TForm1.Button3Click(Sender: TObject);
var i,nn:integer;
s:string; h:double;
begin
  nn:=0;
  for i:= 0 to memo1.Lines.Count-1 do
  nn:=nn+ length(memo1.Lines.Strings[i]);

```

```

h:=360/n;
form1.memo2.Clear;
n:=1;
for i:= 0 to memo1.Lines.Count-1 do
begin
s:=memo1.Lines.Strings[i];
while length(s)>0 do
begin
Mebius(n*h,ABCD(copy(s,1,1))/100-1);
delete(s,1,1);
inc(n);
end; end;

form1.memo2.Lines.SaveToFile('Kod.txt');
end;

procedure TForm1.ColorGrid1Change(Sender: TObject);
begin
checkbox1.Checked:=false;
col2:=ColorGrid1.ForegroundColor;
end;

procedure TForm1.Draw;
var i:integer;
h:double;
begin
glClear(GL_DEPTH_BUFFER_BIT xor GL_COLOR_BUFFER_BIT);
glPolygonMode(GL_FRONT_AND_BACK, {GL_FILL}GL_LINE);
glBegin(GL_TRIANGLES);//рисуем трианглами

for i:=3 to n-1//High(xyz)
do begin
glcolor3f(Col2,Col2 shr 8,Col2 shr 16);

if checkbox1.Checked then glcolor3f(0, 3,1);
glVertex3d(xyz[i][0],xyz[i][1],xyz[i][2]);
if checkbox1.Checked then
glcolor3f(3, 1,0);

glVertex3d(xyz[i-1][0],xyz[i-1][1],xyz[i-1][2]);

glcolor3f(3, 0.1,1);
glVertex3d(xyz[i-2][0],xyz[i-2][1],xyz[i-2][2]);

end;
glEnd();
glDisable(GL_TRIANGLES);
glBegin(GL_LINE_STRIP);
glcolor3f(10, 0,0);
glVertex3f(1, 0,0);
glVertex3f(0, 0,0);

```

```
glColor3f(0, 10,0);
glVertex3f(0, 1,0);
glVertex3f(0, 0,0);

glColor3f(0, 0,10);
glVertex(0, 0, 2);

glEnd;
glRotatef(-0.2, 0,0,1);
SwapBuffers(ghDC);
end;

procedure TForm1.Timer1Timer(Sender: TObject);
begin
draw;
glRotatef(-0.2, 0,0,1);
end;

procedure TForm1.Timer2Timer(Sender: TObject);
begin
timer2.Enabled:=false;
form1.WindowState:=wsmaximized;
end;

procedure TForm1.TrackBar1Change(Sender: TObject);
begin
kr:=TrackBar1.Position;
Form1.Resize;
end;

procedure TForm1.FormClose(Sender: TObject; var Action: TCloseAction);
begin
if ghRC<>0 then
begin
wglMakeCurrent(ghDC,0);
wglDeleteContext(ghRC);
end;
if ghDC<>0 then
ReleaseDC(Handle, ghDC);
end;
end.
```

Код программы в Delphi (декодирование)**Приложение 3**

```
unit Unit1;
interface
uses
```



```

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, ExtCtrls, OpenGL, ComCtrls, StdCtrls, Spin,math, Vcl.ColorGrid;
type
TVector = Array[0..2]of single;//тут храним X,У,З

type
TForm1 = class(TForm)
  Timer1: TTimer;
  Timer2: TTimer;
  Panel1: TPanel;
  Button1: TButton;
  Memo1: TMemo;
  TrackBar1: TTrackBar;
  Button3: TButton;
  Memo2: TMemo;
  CheckBox1: TCheckBox;
  ColorGrid1: TColorGrid;
  ProgressBar1: TProgressBar;
  Label1: TLabel;
  procedure FormCreate(Sender: TObject);
  procedure FormResize(Sender: TObject);
  procedure Timer1Timer(Sender: TObject);
  procedure FormClose(Sender: TObject; var Action: TCloseAction);
  procedure Timer2Timer(Sender: TObject);
  procedure FormDblClick(Sender: TObject);
  procedure Button1Click(Sender: TObject);
  procedure FormDestroy(Sender: TObject);
  procedure TrackBar1Change(Sender: TObject);
  procedure Button3Click(Sender: TObject);
function Mebius(u,v:double):Tvector;
  procedure ColorGrid1Change(Sender: TObject);
private
  ghRC:HGLRC;
  ghDC:HDC;
  procedure Draw;
public
  { Public declarations }
end;
const
a='ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789/*-+
.,~!@#$$%^&()_ =АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯабвгдеёжзийклм
нопрстуфхцчшщъыьэюя';

var
  Form1: TForm1;
xyz0,xyz1,xyz2,d,r,rx,ry,dd,fi:double;
kr,n,nn0:integer;
var
  xyz:array[0..10000000] of TVector;
  col2:tcolor;
implementation

```

```
{SR *.dfm}
function bSetupPixelFormat(DC:HDC):boolean;
var
  pfd:PIXELFORMATDESCRIPTOR;
  ppfd:PPIXELFORMATDESCRIPTOR;
  pixelformat:integer;
begin
  ppfd := @pfd;

  ppfd.nSize := sizeof(PIXELFORMATDESCRIPTOR);
  ppfd.nVersion := 1;
  ppfd.dwFlags := PFD_DRAW_TO_WINDOW xor
    PFD_SUPPORT_OPENGL xor
    PFD_DOUBLEBUFFER;
  ppfd.dwLayerMask := PFD_MAIN_PLANE;
  ppfd.iPixelFormat := PFD_TYPE_RGBA;
  ppfd.cColorBits := 16;
  ppfd.cDepthBits := 16;

  ppfd.cAccumBits := 0;
  ppfd.cStencilBits := 0;

  pixelformat := ChoosePixelFormat(dc, ppfd);
  if pixelformat=0 then
  begin
    MessageBox(0, 'ChoosePixelFormat failed', 'Error', MB_OK);
    bSetupPixelFormat:=FALSE;
    exit;
  end;

  if SetPixelFormat(dc, pixelformat, ppfd)=false then
  begin
    MessageBox(0, 'SetPixelFormat failed', 'Error', MB_OK);
    bSetupPixelFormat:=FALSE;
    exit;
  end;

  bSetupPixelFormat:=TRUE;
end;

procedure TForm1.FormCreate(Sender: TObject);
var
  p: TGLArrayf4;
  d: TGLArrayf3;
n, I: Integer;
begin
  form1.memo1.Lines.LoadFromFile('Kod.txt');

n:=1;
for i:= 0 to memo1.Lines.Count-1 do
```

```

if i mod 4=0 then
begin
xyz[n][0]:=strtofloat(memo1.Lines.Strings[i]);
xyz[n][1]:=strtofloat(memo1.Lines.Strings[i+1]);
xyz[n][2]:=strtofloat(memo1.Lines.Strings[i+2]);
inc(n);
end;
nn0:=trunc(memo1.Lines.Count/4);

label1.Caption:='n = '+inttostr(nn0);
kr:=3;
    dd:=1;
    ghDC := GetDC(Handle);
    if bSetupPixelFormat(ghDC)=false then    Close();
    ghRC := wglCreateContext(ghDC);
    wglMakeCurrent(ghDC, ghRC);
    glClearColor(4.0, 4.0, 4.0, 4.0);
    FormResize(Sender);

    glEnable(GL_COLOR_MATERIAL);
    glEnable(GL_DEPTH_TEST);

    glEnable(GL_LIGHTING);
    glEnable(GL_LIGHT0);
    p[0]:=3;
    p[1]:=3;
    p[2]:=3;
    p[3]:=0;
    d[0]:=0;
    d[1]:=0;
    d[2]:=-3;
    glLightfv(GL_LIGHT0,GL_POSITION,@p);
    glLightfv(GL_LIGHT0,GL_SPOT_DIRECTION,@d);

    glEnable(GL_LIGHT1);
    p[0]:=-3;
    p[1]:=-3;
    p[2]:=3;
    p[3]:=0;
    d[0]:=0;
    d[1]:=0;
    d[2]:=-3;
    glLightfv(GL_LIGHT1,GL_POSITION,@p);
    glLightfv(GL_LIGHT1,GL_SPOT_DIRECTION,@d);

    end;

procedure TForm1.FormDbClick(Sender: TObject);
begin
application.Terminate;
end;

```

```
procedure TForm1.FormResize(Sender: TObject);
begin
  glViewport( 0, 0, Width-100, Height );
  glMatrixMode( GL_PROJECTION );
  glLoadIdentity();
  glOrtho(-kr/2,kr/2, -kr/2,kr/2,  0,30);

  gluLookAt( 2,2,2,  0,0,0,  0,0,10);
  glMatrixMode( GL_MODELVIEW );
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
  Timer1.Enabled:=false;
  Button3.Click;
  Timer1.Enabled:=true;
end;

function TForm1.Mebius(u,v:double):Tvector;
begin
  xyz0:=(1+(v/2)*cos(u/2*pi/180))*cos(u*pi/180);
  xyz1:=(1+(v/2)*cos(u/2*pi/180))*sin(u*pi/180);
  xyz2:=(v/2)*sin(u/2*pi/180);
end;

function ABCD(s:string):integer;
var i:integer;
begin
  for i:=1 to length(a) do
  if s=a[i] then
  begin
  ABCD:=i;
  exit;
  end;
end;

procedure TForm1.Button3Click(Sender: TObject);
var j,i,nn:integer;
mini,s:string; d0,d, h:double;
begin
  n:=1;
  for i:= 0 to memo1.Lines.Count-1 do
  if i mod 4=0 then
  begin
  xyz[n][0]:=strtofloat(memo1.Lines.Strings[i]);
  xyz[n][1]:=strtofloat(memo1.Lines.Strings[i+1]);
  xyz[n][2]:=strtofloat(memo1.Lines.Strings[i+2]);
  inc(n);
  end;
end;
```

```

nn:=trunc(memo1.Lines.Count/4);
progressbar1.Max:=nn;
h:=360/nn;
label1.Caption:='n = '+inttostr(nn);

form1.memo2.Clear;

for i:= 1 to nn do
begin
progressbar1.Position:=i;
d0:=10;
for j:= 1 to length(a) do
begin
Mebius(i*h,j/100-1);
d:= sqrt(sqr(xyz0-xyz[i][0])+sqr(xyz1-xyz[i][1])+sqr(xyz2-xyz[i][2]));
if d<=d0 then begin d0:=d; mini:=a[j];end;
end;
memo2.Text:=memo2.Text+mini;
end;
form1.memo2.Lines.SaveToFile('text.txt');
end;

procedure TForm1.ColorGrid1Change(Sender: TObject);
begin
checkbox1.Checked:=false;
col2:=ColorGrid1.ForegroundColor;
end;
procedure TForm1.Draw;
var i:integer;
h:double;
begin
glClear(GL_DEPTH_BUFFER_BIT xor GL_COLOR_BUFFER_BIT);
glPolygonMode(GL_FRONT_AND_BACK, {GL_FILL}GL_LINE);
glBegin(GL_TRIANGLES);//рисуем треугольниками
for i:=3 to nn0-1//High(xyz)
do begin
glcolor3f(Col2,Col2 shr 8,Col2 shr 16);
if checkbox1.Checked then glColor3f(0, 3,1);
glVertex3d(xyz[i][0],xyz[i][1],xyz[i][2]);
if checkbox1.Checked then
glcolor3f(3, 1,0);
glVertex3d(xyz[i-1][0],xyz[i-1][1],xyz[i-1][2]);
glcolor3f(3, 0.1,1);
glVertex3d(xyz[i-2][0],xyz[i-2][1],xyz[i-2][2]);
end;
glEnd();
glDisable(GL_TRIANGLES);
glBegin(GL_LINE_STRIP);
glcolor3f(10, 0,0);
glVertex3f(1, 0,0);

```



```
glVertex3f(0, 0,0);

glColor3f(0, 10,0);
glVertex3f(0, 1,0);
glVertex3f(0, 0,0);

glColor3f(0, 0,10);
glVertex(0, 0, 2);

glEnd;
glRotatef(-0.2, 0,0,1);
SwapBuffers(ghDC);
end;

procedure TForm1.Timer1Timer(Sender: TObject);
begin
draw;
glRotatef(-0.2, 0,0,1);
end;

procedure TForm1.Timer2Timer(Sender: TObject);
begin
timer2.Enabled:=false;
form1.WindowState:=wsmaximized;
end;

procedure TForm1.TrackBar1Change(Sender: TObject);
begin
kr:=TrackBar1.Position;
Form1.Resize;
end;

procedure TForm1.FormClose(Sender: TObject; var Action: TCloseAction);
begin
if ghRC<>0 then
begin
wglMakeCurrent(ghDC,0);
wglDeleteContext(ghRC);
end;
if ghDC<>0 then
ReleaseDC(Handle, ghDC);
end;
end.
```