8. Поиск в глубину. Эйлеров цикл. Гамильтонов цикл.

***Определение 1.*** *Эйлеровым путем в*[*графе*](https://lms2.sseu.ru/courses/eresmat/gloss/g279.htm)*называется*[*путь*](https://lms2.sseu.ru/courses/eresmat/gloss/g289.htm)*, содержащий все ребра графа и проходящий через каждое по одному разу.*

Пример 1. Рассмотрим граф

|  |
| --- |
|  |
|  | https://lms2.sseu.ru/courses/eresmat/course2/razd5_2/par5_7k2.files/image003.gif |

Он имеет эйлеров путь (*x*4,*x*1,*x*3,*x*2,*x*1,*x*5,*x*3).

***Определение 2.*** *Эйлеровым циклом в графе называется*[*цикл*](https://lms2.sseu.ru/courses/eresmat/gloss/g291.htm)*, содержащий все ребра графа и проходящий через каждое по одному разу.*

***Определение 3.***  *Граф, обладающий эйлеровым циклом, называется эйлеровым графом.*

Пример 2. Рассмотрим граф

|  |
| --- |
|  |
|  | https://lms2.sseu.ru/courses/eresmat/course2/razd5_2/par5_7k2.files/image004.gif |

Данный граф является эйлеровым, так как он имеет эйлеров цикл (*x*2,*x*5,*x*4,*x*1,*x*2,*x*3,*x*4,*x*2).

**Теорема 1.** Эйлеров [граф связный](https://lms2.sseu.ru/courses/eresmat/gloss/g293.htm), и все его вершины четны.

*Доказательство.*

Связность следует из определения эйлерового графа. Эйлеров цикл содержит каждое ребро и притом только один раз. Следовательно, степень каждой вершины графа должна состоять из двух одинаковых слагаемых: количество входов в вершину и количество выходов из вершины.

**Теорема 2.** Если граф  *G(X,T)*  связный и все его вершины четны, то он обладает эйлеровым циклом.

**Теорема 3.** Если граф  *G(X,T)*  обладает эйлеровым путем с концами *А* и *В*, то граф  *G(X,T)*  связный и *А* и *В* его единственные нечетные вершины.

*Доказательство.*

Если путь начинается в *А* и кончается в *В*, то *А* и *В* нечетные вершины, даже если путь неоднократно проходил через них. В любую другую вершину путь должен привести и вывести из нее, т.е. остальные вершины четные.

**Теорема 4.** Если граф  *G(X,T)*  связный и *А* и *В* его единственные нечетные вершины, то граф обладает эйлеровым путем с концами *А* и *В*.

**Теорема 5.** Если граф  *G(X,T)*  связный, то можно построить цикличный маршрут, содержащий все ребра в точности 2 раза, по одному в каждом направлении.

**{Фигня которая мб понадобится}**

Эйлеровы циклы

Эйлеровым циклом (путем) называется цикл (путь), проходящий через все ребра графа. Граф, в котором имеется эйлеров цикл, называют эйлеровым графом.

Теорема об эйлеровом цикле. Связный граф эйлеров тогда и только тогда, когда в нем степени всех вершин четны.

Следствие (об эйлеровом пути). Эйлеров путь в связном графе существует тогда и только тогда, когда в нем имеется не более двух вершин нечетной степени.

Ниже приводится алгоритм построения эйлерова цикла. Предполагается, что условия существования уже проверены – граф связен и степени четны. В алгоритме строится путь, начинающийся в произвольно выбранной стартовой вершине a, при этом каждый раз для дальнейшего продвижения выбирается любое еще не пройденное ребро. Вершины пути накапливаются в стеке S (они могут повторяться). Когда наступает тупик – все ребра, инцидентные последней вершине пути, уже пройдены, производится возвращение вдоль пройденного пути, пока не встретится вершина, которой инцидентно не пройденное ребро. При возвращении вершины перекладываются из стека S в другой стек C. Затем возобновляется движение вперед по не пройденным ребрам, пока снова не наступает тупик, и т. д. Процесс заканчивается, когда стек S оказывается пустым. В этот момент в стеке C находится последовательность вершин эйлерова цикла.

Построение эйлерова цикла

1 выбрать произвольно вершину а;

2 a⇒S;

3 while S ≠ ∅ do

4 x := top ( S ) ;

5 if имеется не пройденное ребро ( x, y )

6 then пометить ребро ( x, y ) как пройденное;

7 y⇒S;

8 else переместить вершину x из S в C

Трудоемкость этого алгоритма оценивается как O(m) . Этот вывод справедлив лишь

при определенных предположениях о том, как задан граф.

Можно, например, использовать две структуры:

• массив ребер, в котором в позиции, соответствующей ребру, помещается запись,

указывающая две вершины этого ребра и пометку о том, пройдено ли это ребро;

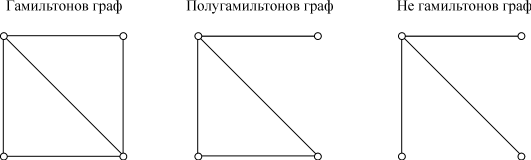
• списки инцидентности, в которых для каждой вершины перечисляются

инцидентные ей ребра.

В ориентированном графе эйлеров цикл – это ориентированный цикл, проходящий через все ребра.

**Гамильтоновый цикл:**

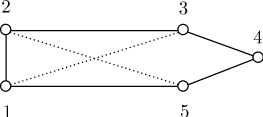
Если граф имеет простой цикл, содержащий все вершины графа по одному разу, то такой цикл называется гамильтоновым циклом, а граф называется гамильтоновым графом. Граф, который содержит простой путь, проходящий через каждую его вершину, называется полугамильтоновым. Это определение можно распространить на ориентированные графы, если путь считать ориентированным.

[](https://sites.google.com/a/labore.ru/teoria-grafov-i-ee-primenenie/vvedenie-v-teoriu/gamiltonov-cikl-i-gamiltonov-graf/5_1.gif?attredirects=0)

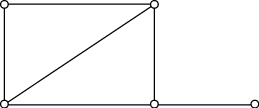
Гамильтонов цикл не обязательно содержит все ребра графа. Ясно, что гамильтоновым может быть только связный граф и, что всякий гамильтонов граф является полугамильтоновым. Заметим, что гамильтонов цикл существует далеко не в каждом графе.

Рассмотрим несколько достаточных условий существования гамильтоновых циклов в графе.  
  
Во-первых, всякий полный граф является гамильтоновым. Действительно, он содержит такой простой цикл, которому принадлежат все вершины данного графа.

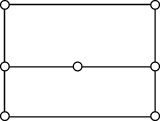
Во-вторых, если граф, помимо простого цикла, проходящего через все его вершины, содержит и другие ребра, то он также является гамильтоновым.

[](https://sites.google.com/a/labore.ru/teoria-grafov-i-ee-primenenie/vvedenie-v-teoriu/gamiltonov-cikl-i-gamiltonov-graf/5_2.gif?attredirects=0)

Простой (гамильтонов) цикл выделен сплошной линией (1, 2), (2, 3), (3, 4), (4, 5), (5, 1). Заметим, что если граф имеет один гамильтонов цикл, то он может иметь и другие гамильтоновы циклы.  
  
Если гамильтонов граф объединить с еще одной вершиной ребром так, что образуется висячая вершина, то такой граф гамильтоновым не является, поскольку не содержит простого цикла, проходящего через все вершины графа.

[](https://sites.google.com/a/labore.ru/teoria-grafov-i-ee-primenenie/vvedenie-v-teoriu/gamiltonov-cikl-i-gamiltonov-graf/5_3.gif?attredirects=0)

Не является гамильтоновым и граф, представляющий собой простой цикл с "перекладиной", на которой расположены одна или несколько вершин.

[](https://sites.google.com/a/labore.ru/teoria-grafov-i-ee-primenenie/vvedenie-v-teoriu/gamiltonov-cikl-i-gamiltonov-graf/5_4.gif?attredirects=0)

**{Более сложное обьяснение}**

Гамильтоновы пути и циклы:

Гамильтоновым циклом (путем) называют простой цикл (путь), содержащий все вершины графа. В графе, изображенном на [рис. 8.1](https://intuit.ru/studies/courses/101/101/lecture/2957?page=2" \l "image.8.1) слева, гамильтоновым циклом является, например, последовательность 1, 2, 3, 5, 4, 1. В графе, изображенном в центре, нет гамильтоновых циклов, но есть гамильтоновы пути, например, 2,1,3,5,4. В правом графе нет и гамильтоновых путей.

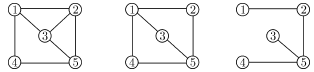


Рис. 8.1.

**Внешне****определение гамильтонова****цикла похоже** на определение эйлерова цикла. Однако есть кардинальное различие в сложности решения соответствующих задач распознавания и построения. Мы видели, что имеется достаточно простой критерий существования эйлерова цикла и эффективный алгоритм его построения. Для гамильтоновых же циклов (и путей) неизвестно никаких просто проверяемых необходимых и достаточных условий их существования, а все известные алгоритмы требуют для некоторых графов перебора большого числа вариантов.

Гамильтонов цикл представляет собой, с комбинаторной точки зрения, просто перестановку вершин графа. При этом в качестве начальной вершины цикла можно выбрать любую вершину, так что можно рассматривать перестановки с фиксированным первым элементом. Самый бесхитростный план поиска гамильтонова цикла состоит в последовательном рассмотрении всех этих перестановок и проверке для каждой из них, представляет ли она цикл в данном графе. Такой способ действий уже при не очень большом числе вершин становится практически неосуществимым ввиду быстрого роста числа перестановок - имеется (n-1)! перестановок из n элементов с фиксированным первым элементом.

Более рациональный подход состоит в рассмотрении всевозможных простых путей, начинающихся в произвольно выбранной стартовой вершине a, до тех пор, пока не будет обнаружен гамильтонов цикл или все возможные пути не будут исследованы. По сути дела, речь тоже идет о переборе перестановок, но значительно сокращенном - если, например, вершина b не смежна с вершиной a, то все (n-2)! перестановок, у которых на первом месте стоит a, а на втором b, не рассматриваются.

Рассмотрим этот алгоритм подробнее. Будем считать, что граф задан окрестностями вершин: для каждой вершины x задано множество вершин, смежных с x. На каждом шаге алгоритма имеется уже построенный отрезок пути, он хранится в стеке PATH. Для каждой вершины x, входящей в PATH, хранится множество N(x) всех вершин, смежных с x, которые еще не рассматривались в качестве возможных продолжений пути из вершины x. Когда вершина x добавляется к пути, множество N(x) полагается равным V(x). В дальнейшем рассмотренные вершины удаляются из этого множества. Очередной шаг состоит в исследовании окрестности последней вершины x пути PATH. Если N(x)\ne \varnothing и в N(x) имеются вершины, не принадлежащие пути, то одна из таких вершин добавляется к пути. В противном случае вершина x исключается из стека. Когда после добавления к пути очередной вершины оказывается, что путь содержит все вершины графа, остается проверить, смежны ли первая и последняя вершины пути, и при утвердительном ответе выдать очередной гамильтонов цикл.

**Алгоритм 2.** Поиск гамильтоновых циклов

1. выбрать произвольно вершину a
2. a\Rightarrow PATH
3. N(a):=V(a)
4. while PATH\ne
   \varnothing do
5. x\, :={\rm top}(PATH)
6. if N(x)\ne \varnothing
7. then взять y\in
   N(x)
8. N(x):=N(x)-y
9. if вершина y не находится в PATH
10. then y\Rightarrow PATH
11. N(y):=V(y)
12. if PATH содержит все вершины
13. then if y смежна с a
14. then выдать цикл
15. else удалить вершину x из PATH

Этот алгоритм очень похож на алгоритм поиска в глубину и отличается от него по существу только тем, что открытая вершина, когда вся ее окрестность исследована, не закрывается, а опять становится новой (исключается из стека). В начале все вершины новые. Процесс заканчивается, когда все вершины опять станут новыми. На самом деле это и есть поиск в глубину, только не в самом графе, а в дереве путей. Вершинами этого дерева являются всевозможные простые пути, начинающиеся в вершине a, а ребро дерева соединяет два пути, один из которых получается из другого добавлением одной вершины в конце. На [рис. 8.2](https://intuit.ru/studies/courses/101/101/lecture/2957?page=2" \l "image.8.2) показаны граф и его дерево путей из вершины 1.

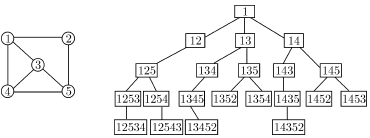


Рис. 8.2.