

ВВЕДЕНИЕ В ТЕОРИЮ ГРАФОВ

Желобаев А.Л. МИЭТ 2009.

1. Основные понятия теории графов

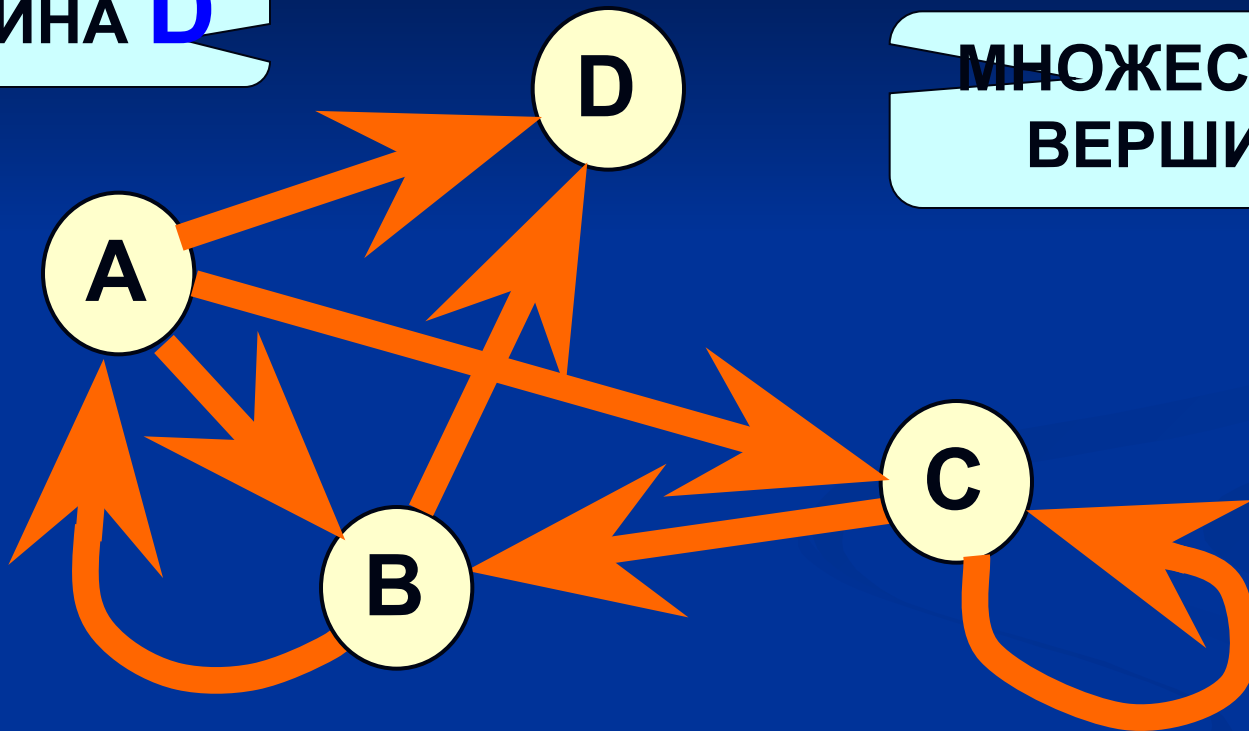
Ориентированный Граф

$G(V,E)$

ВЕРШИНА **D**

МНОЖЕСТВО
ДУГ

МНОЖЕСТВО
ВЕРШИН



ДУГА **{A,B}**

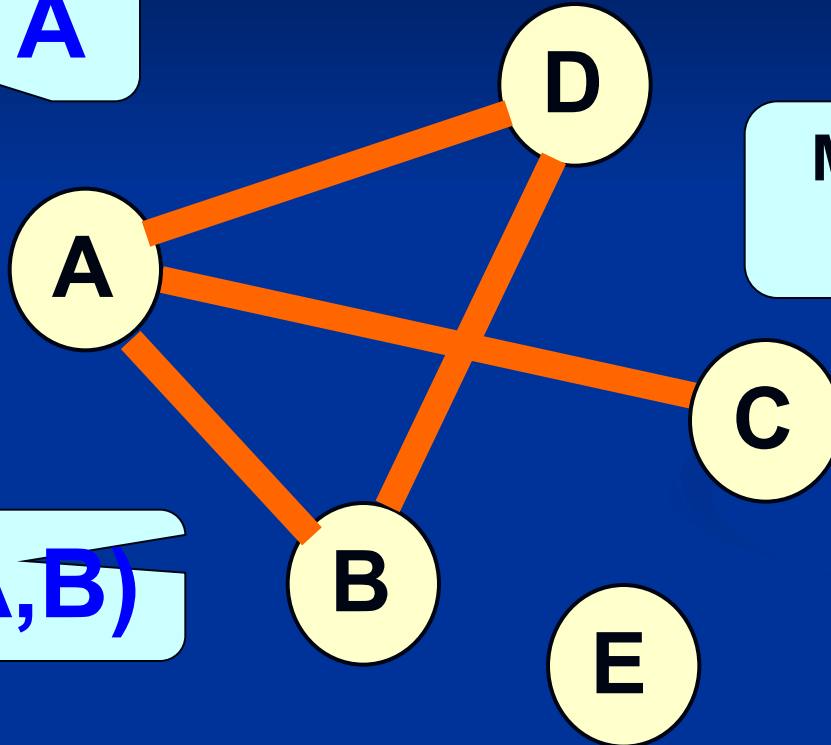
ДУГА **{B,A}**

ЦИКЛ
(Петля)

Неориентированный Граф $G(V,E)$

ВЕРШИНА **A**

МНОЖЕСТВО
РЕБЕР



МНОЖЕСТВО
ВЕРШИН

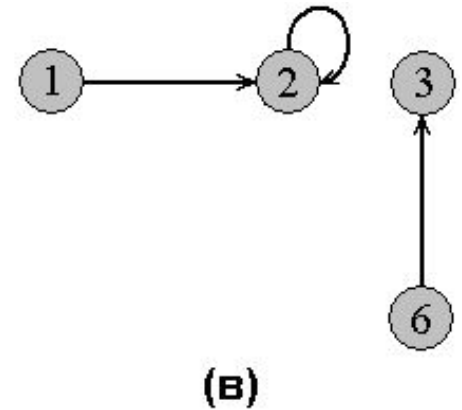
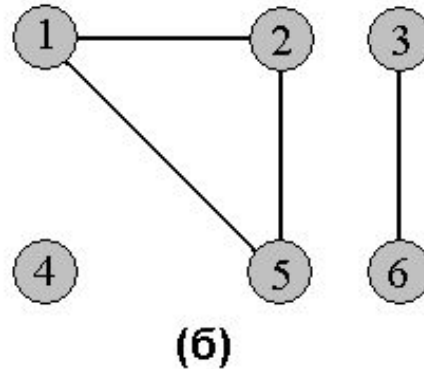
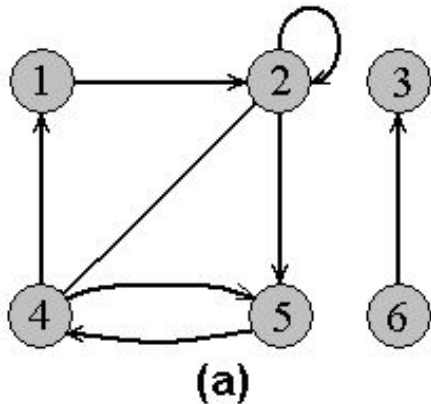
РЕБРО **(A,B)**

ВИСЯЧАЯ
ВЕРШИНА

$(B,A) = (A,B)$

ИЗОЛИРОВАННАЯ
ВЕРШИНА

Ориентированные и неориентированные графы



Ориентированный граф $G(V,E)$,

$$V = \{1,2,3,4,5,6\}$$

$$E = \{\{1,2\}, \{2,2\}, \{2,4\}, \{2,5\}, \{4,1\}, \{4,5\}, \{5,4\}, \{6,3\}\}$$

Неориентированный граф $G(V,E)$,

$$V = \{1,2,3,4,5,6\}$$

$$E = \{(1,2), (1,5), (2,5), (3,6)\}$$

Подграф графа (а), порожденный множеством вершин $\{1,2,3,6\}$

Основные понятия

Вершина графа

- Смежная
- Изолированная
- Висячая
- Степень вершины
исходящая,
входящая

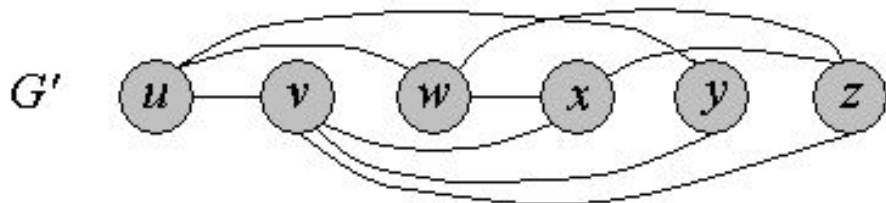
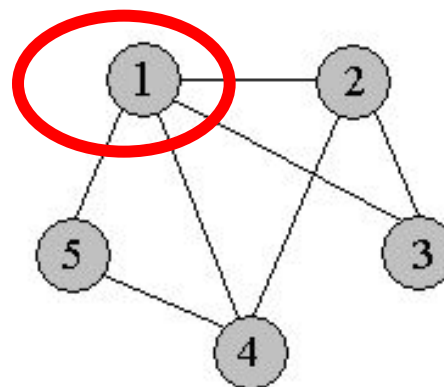
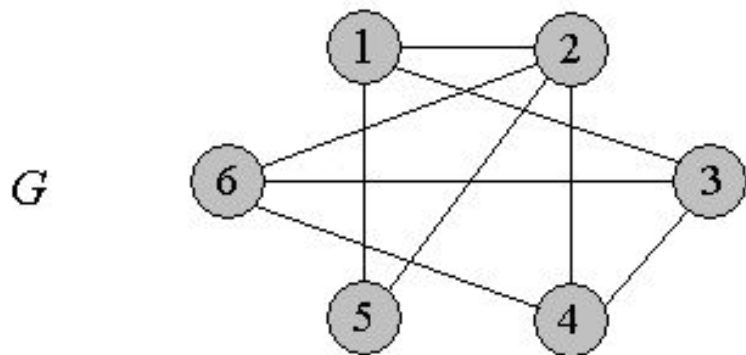
Ребро (дуга) графа

- Инцидентность
- вес
- Дуга-цикл

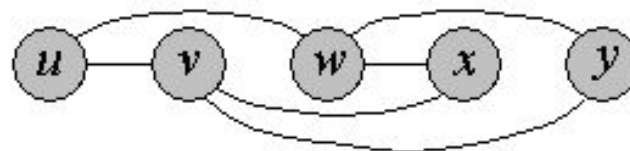
Совокупность дуг

- Путь длины k
- Цикл
- Ациклический граф
- Связный граф
- Сильно связный граф
- Полный граф
- Пустой граф
- Лес
- Дерево в графе

Изоморфизм графов



(a)



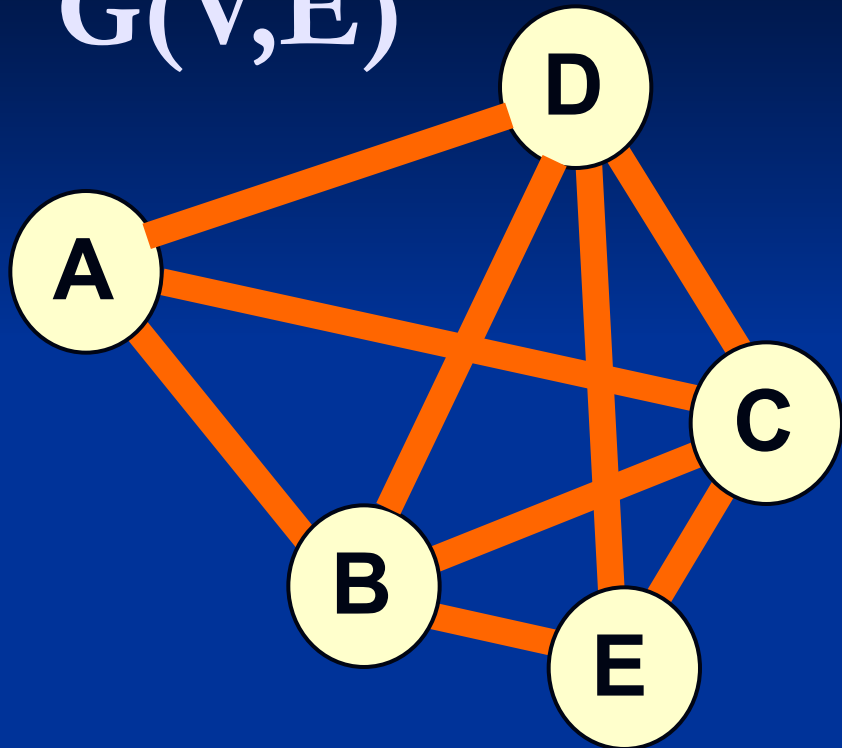
(б)

ИЗОМОРФНЫЕ ГРАФЫ

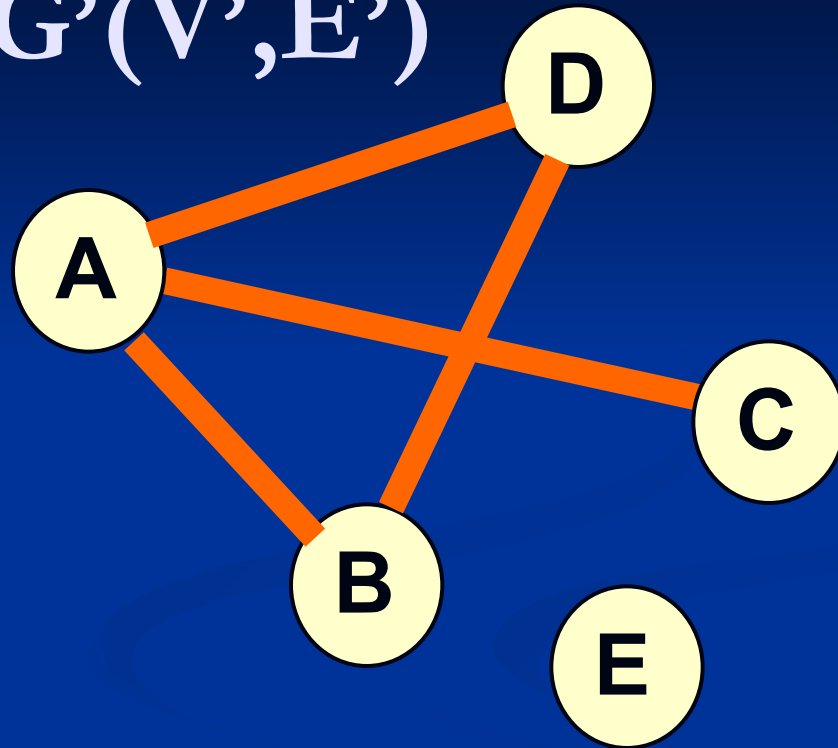
НЕИЗОМОРФНЫЕ ГРАФЫ

Подграфы

$G(V, E)$



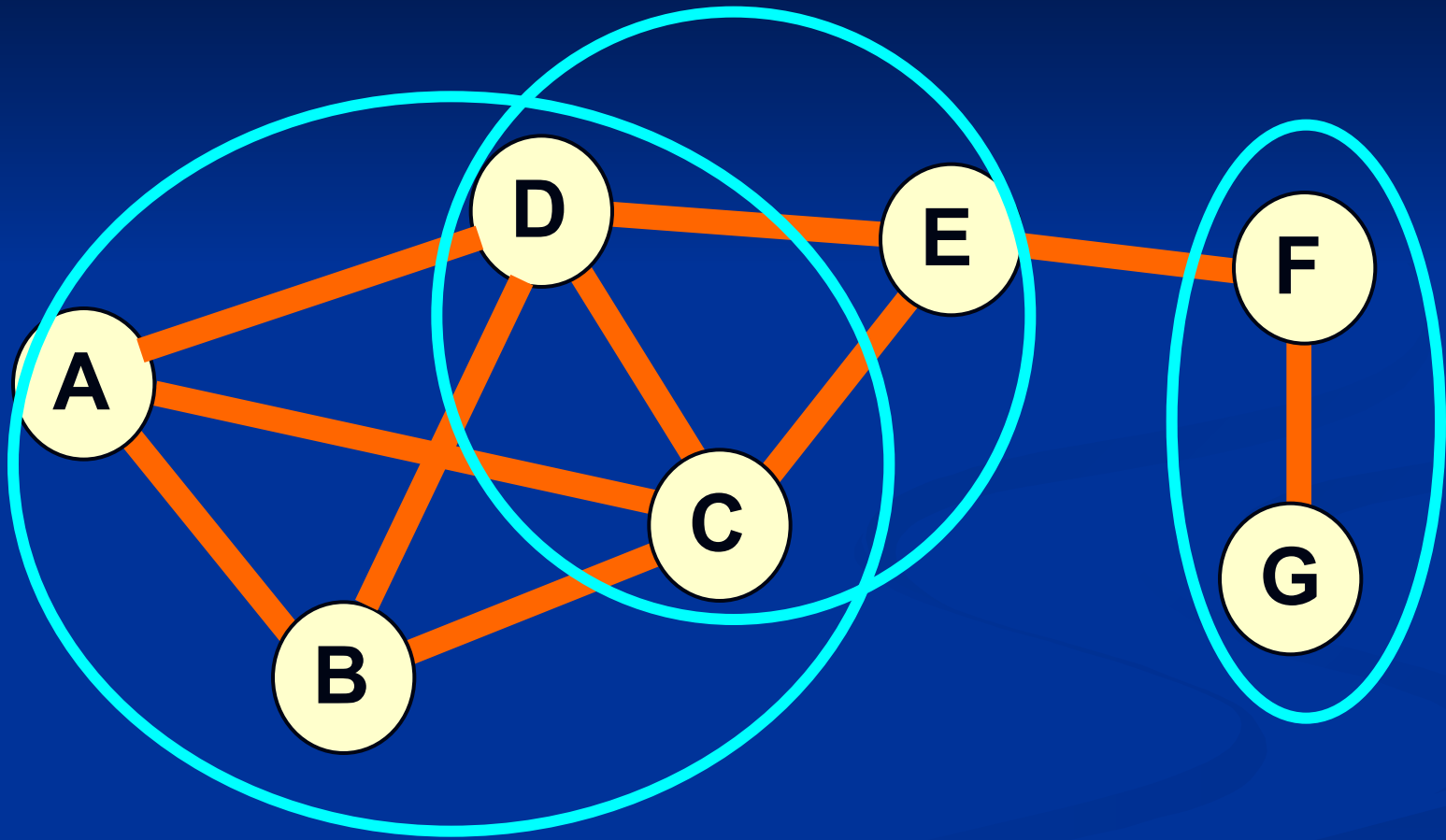
$G'(V', E')$



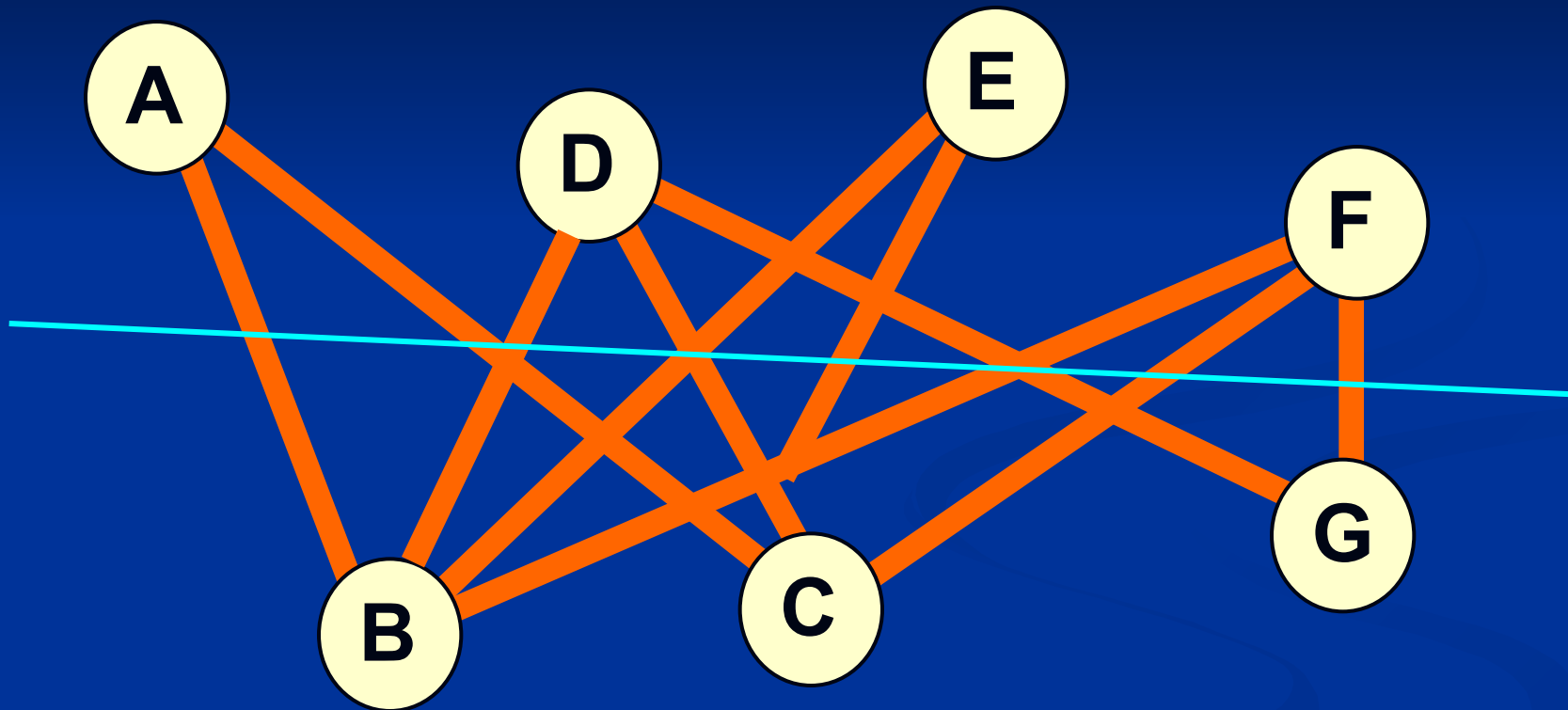
G' подграф G , если $E' \subseteq E$ и $V' \subseteq V$

G' суграф G , если $E' \subseteq E$ и $V' = V$

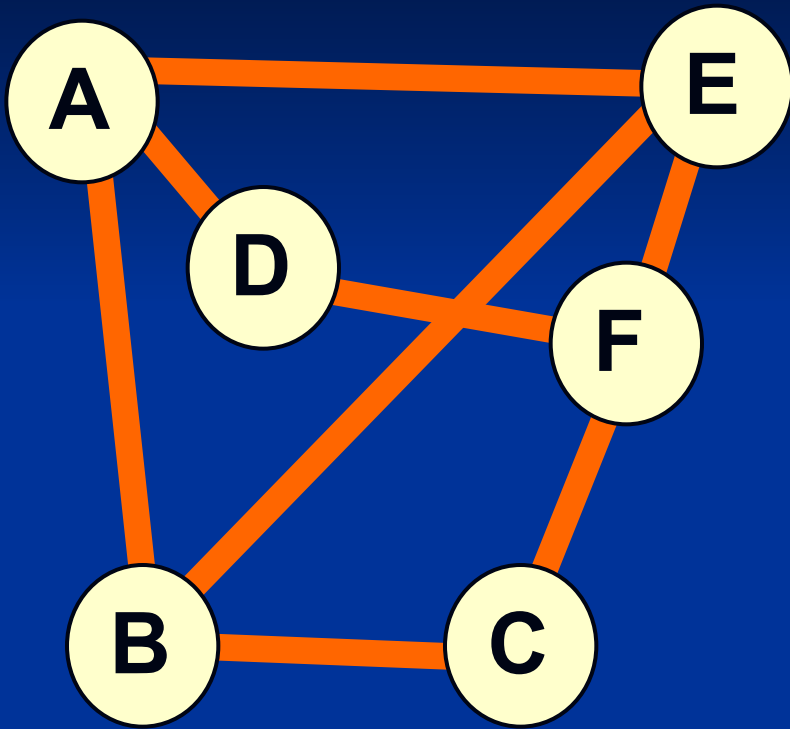
Клики в графе



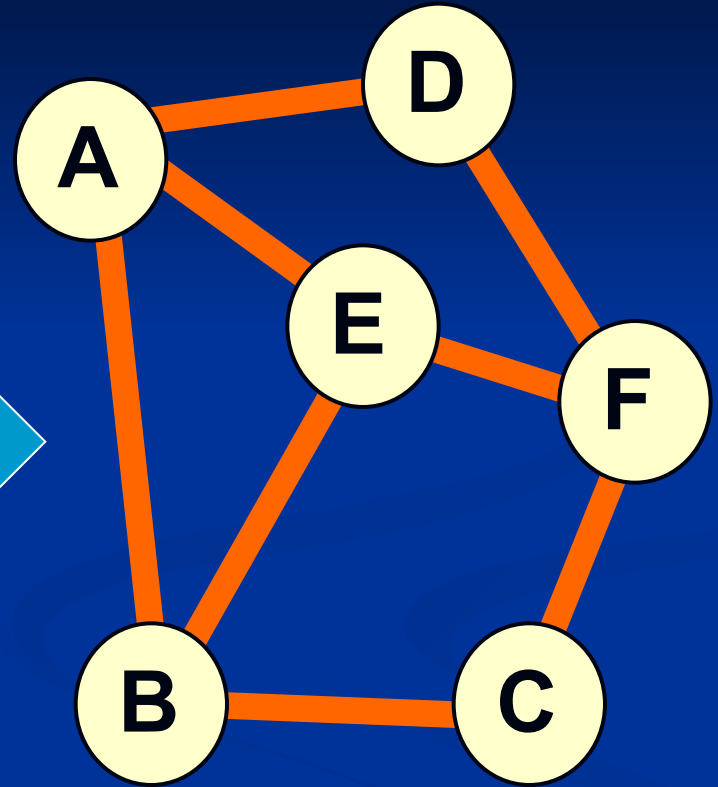
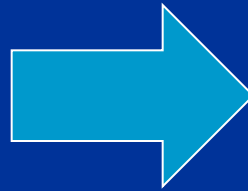
Двудольные графы



Планарные и плоские графы



Планарный
граф



Плоский
граф

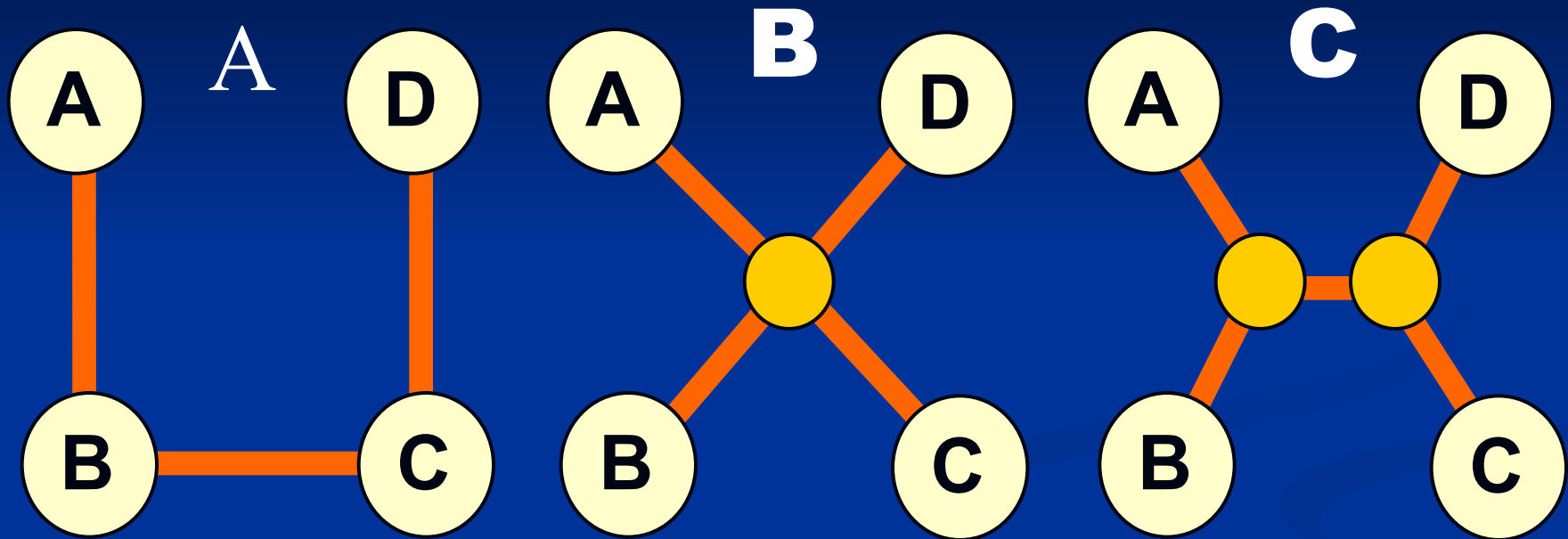
2. Алгоритмы на графах

Минимальные покрывающие деревья

- Имеется граф $G(V,E)$
- Каждому ребру (u,v) задан неотрицательный вес $w(u,v)$
- Задача: найти подмножество $T \subseteq E$, связывающее все вершины, для которого минимален суммарный вес

$$w(T) = \sum_{(u,v) \in T} w(u,v)$$

Отличия теории и практики



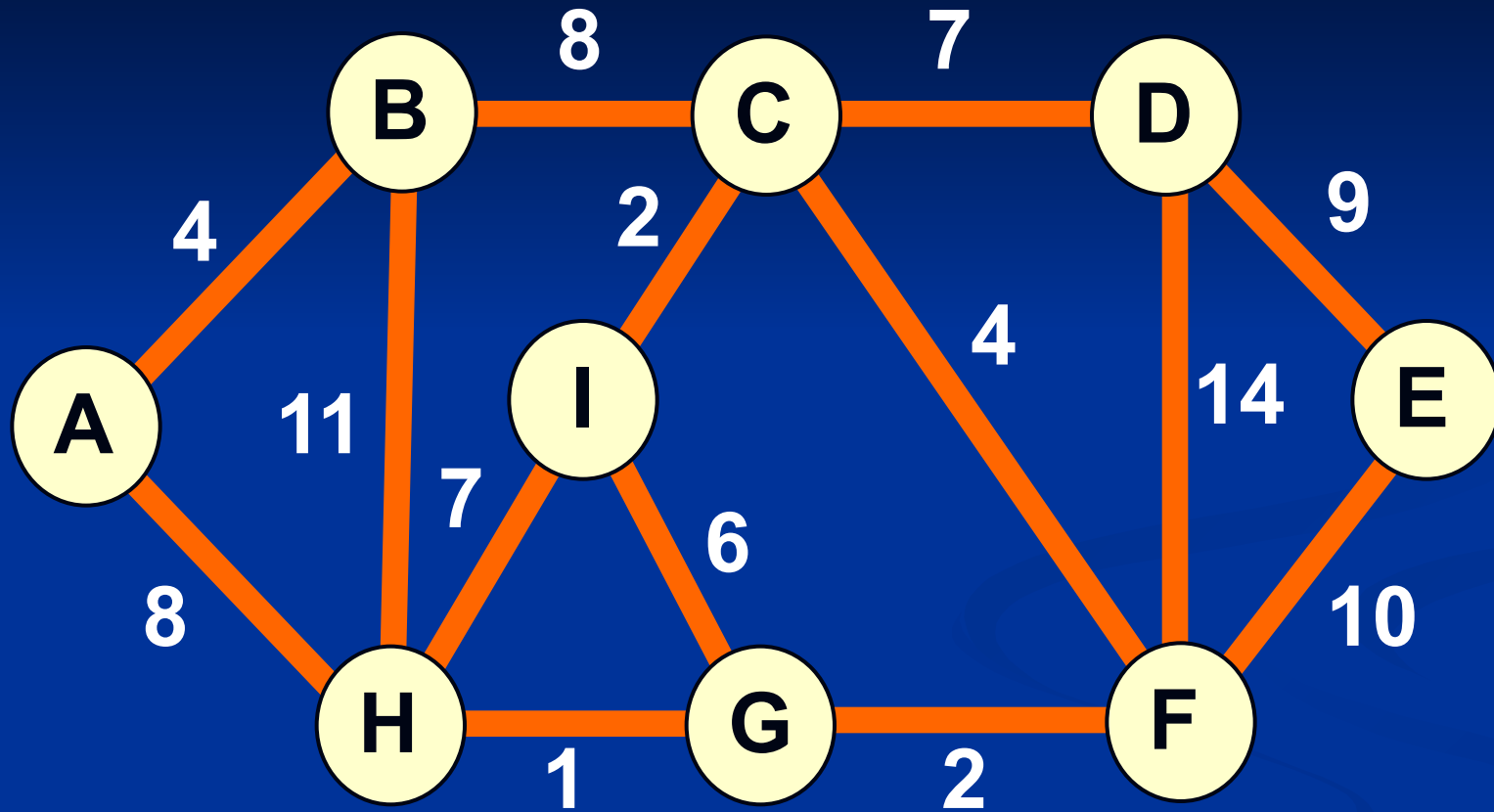
кратчайшее дерево:

A - без дополнительных вершин

B - с дополнительной вершиной

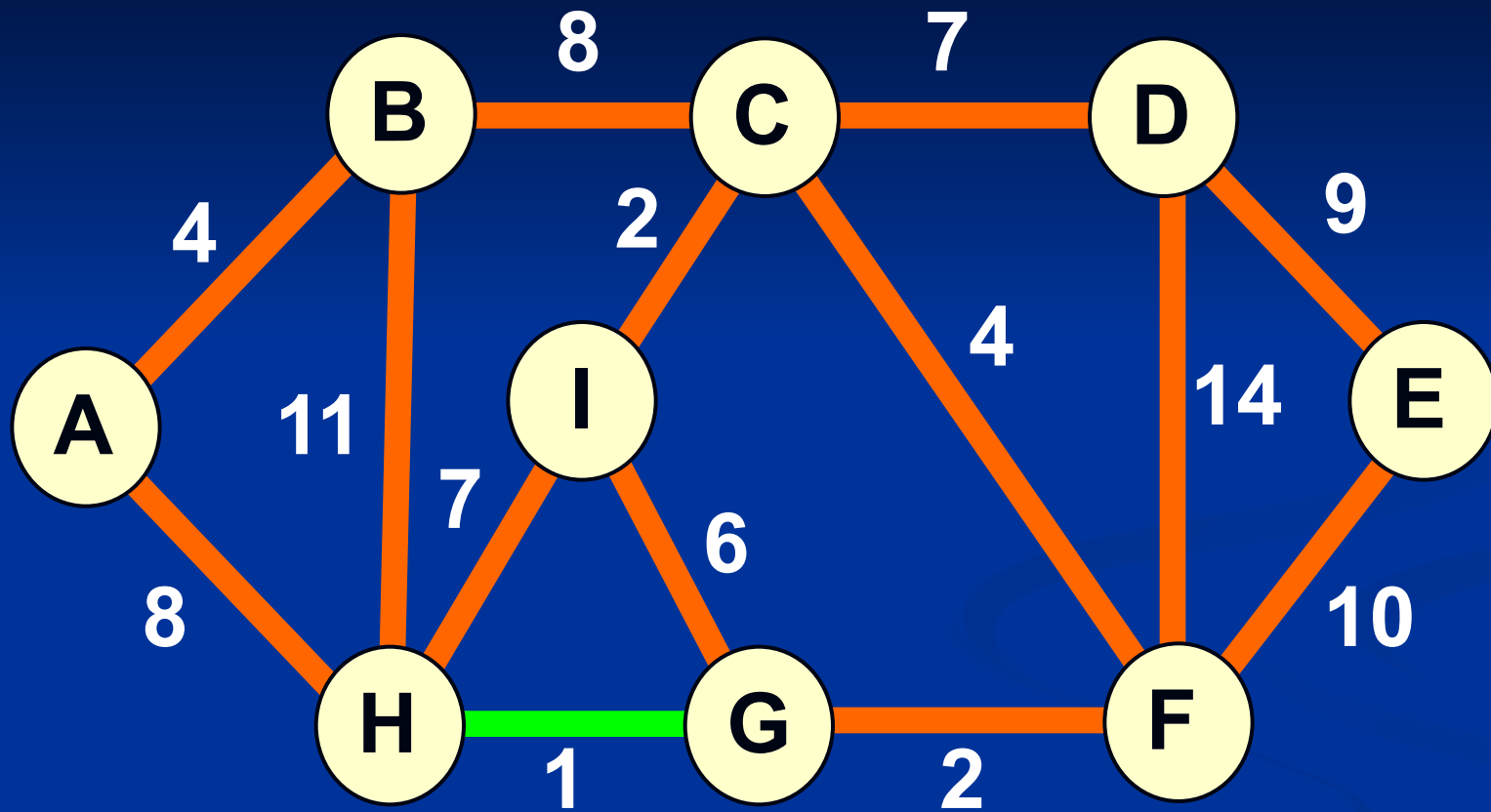
C - дерево Штейнера

Алгоритм Краскала шаг 0



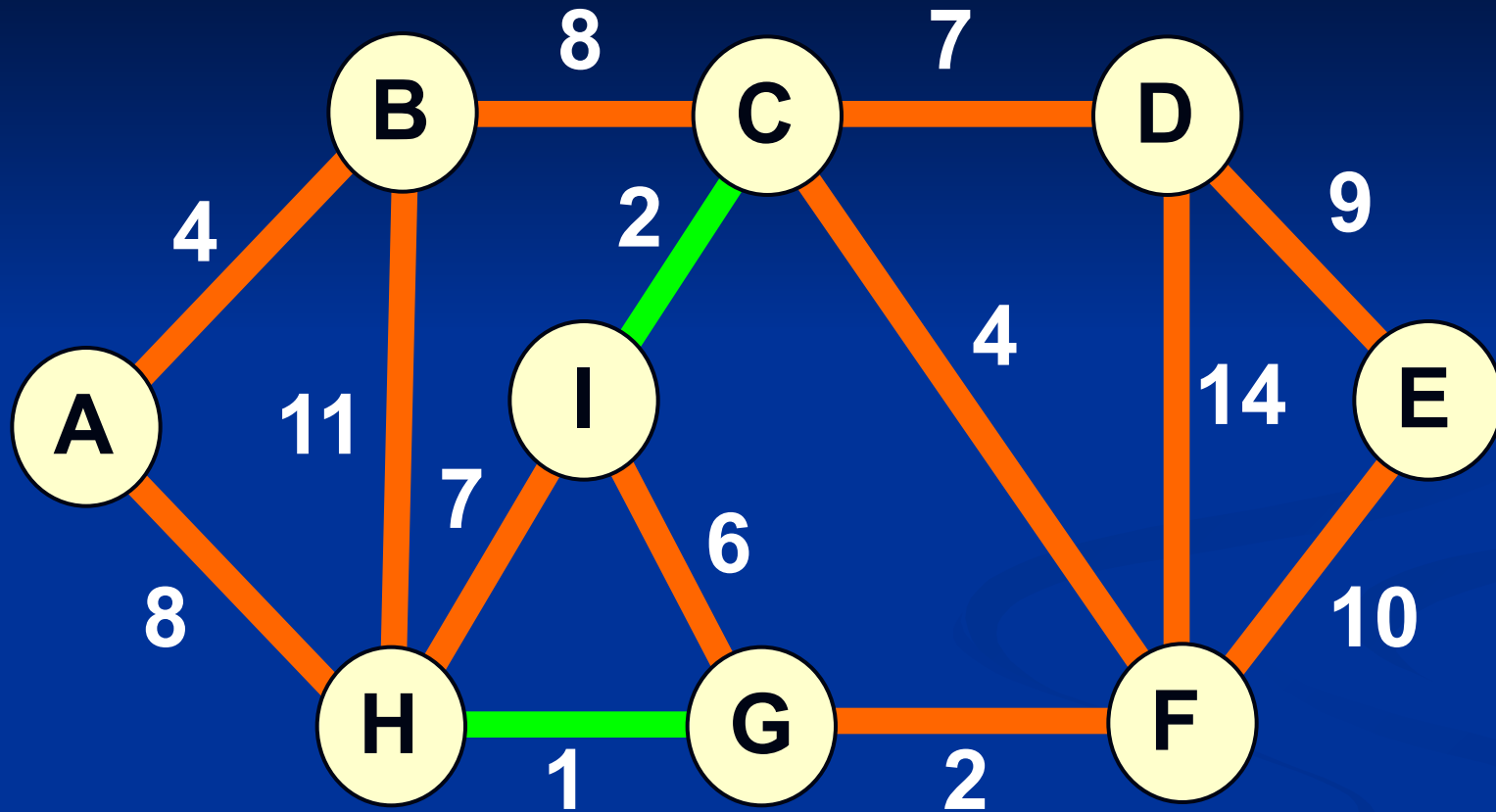
- Суммарная длина деревьев = 0

Алгоритм Краскала шаг 1



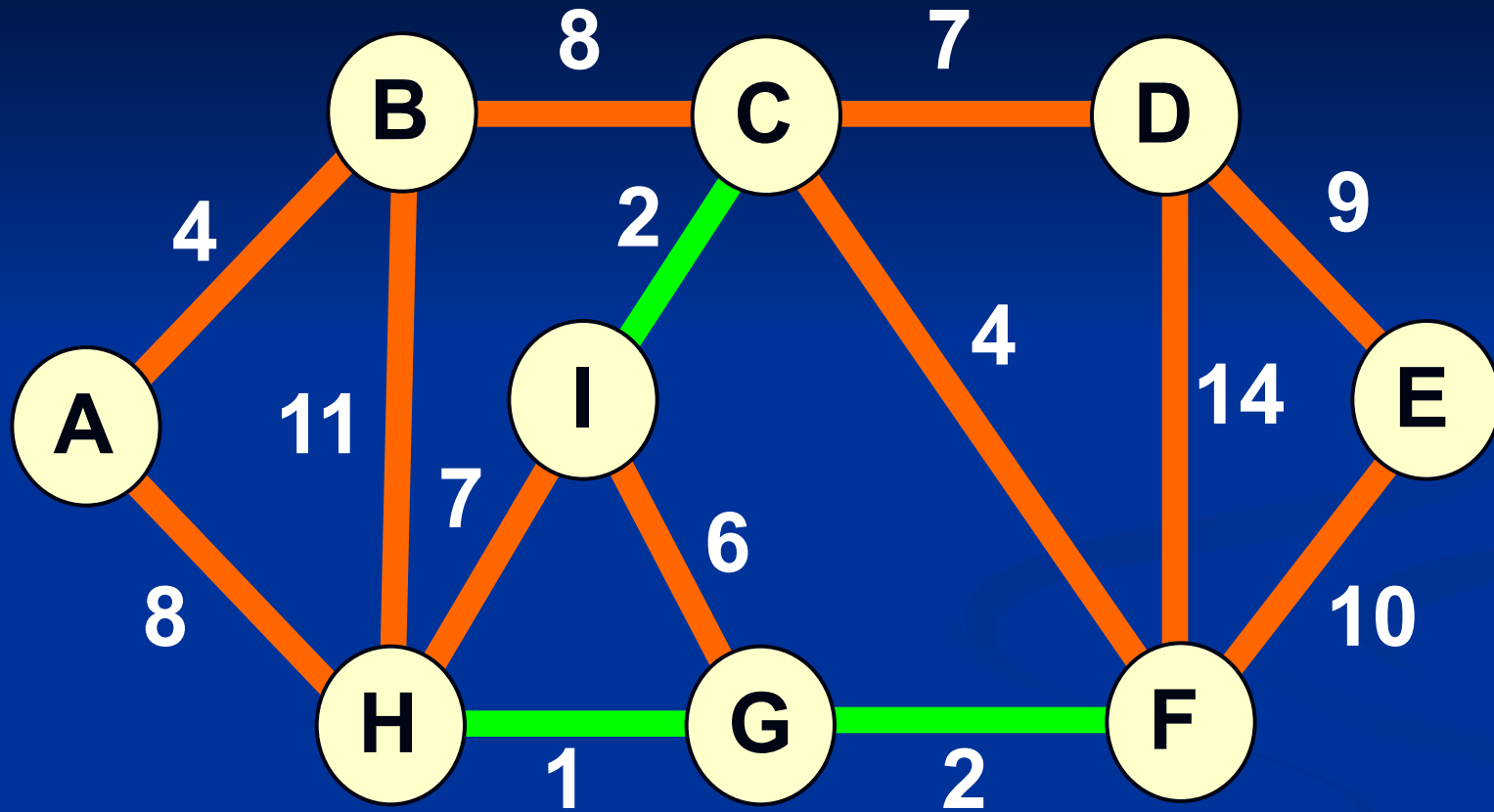
- Суммарная длина деревьев = 1

Алгоритм Краскала шаг 2



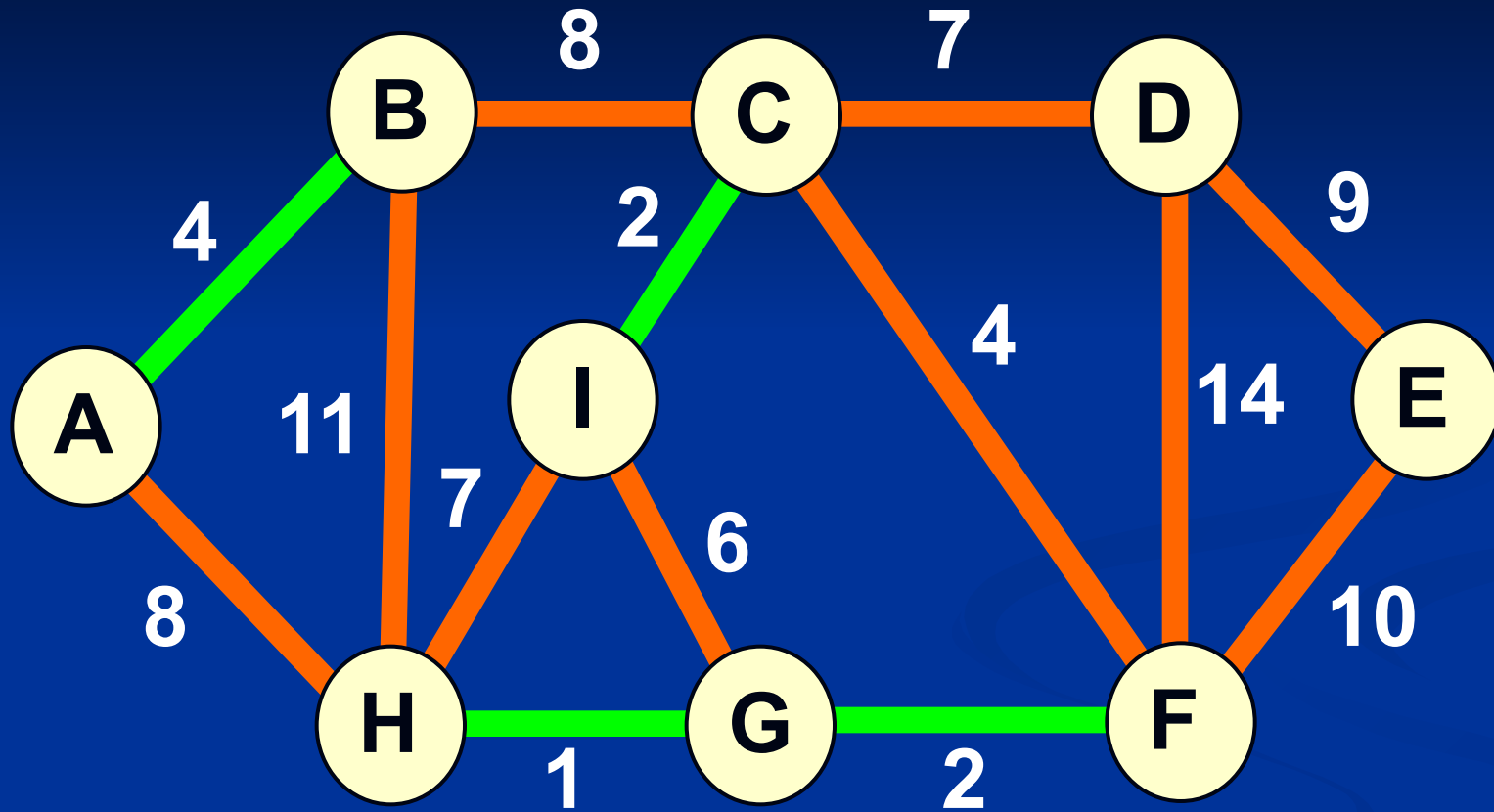
- Суммарная длина деревьев = 3

Алгоритм Краскала шаг 3



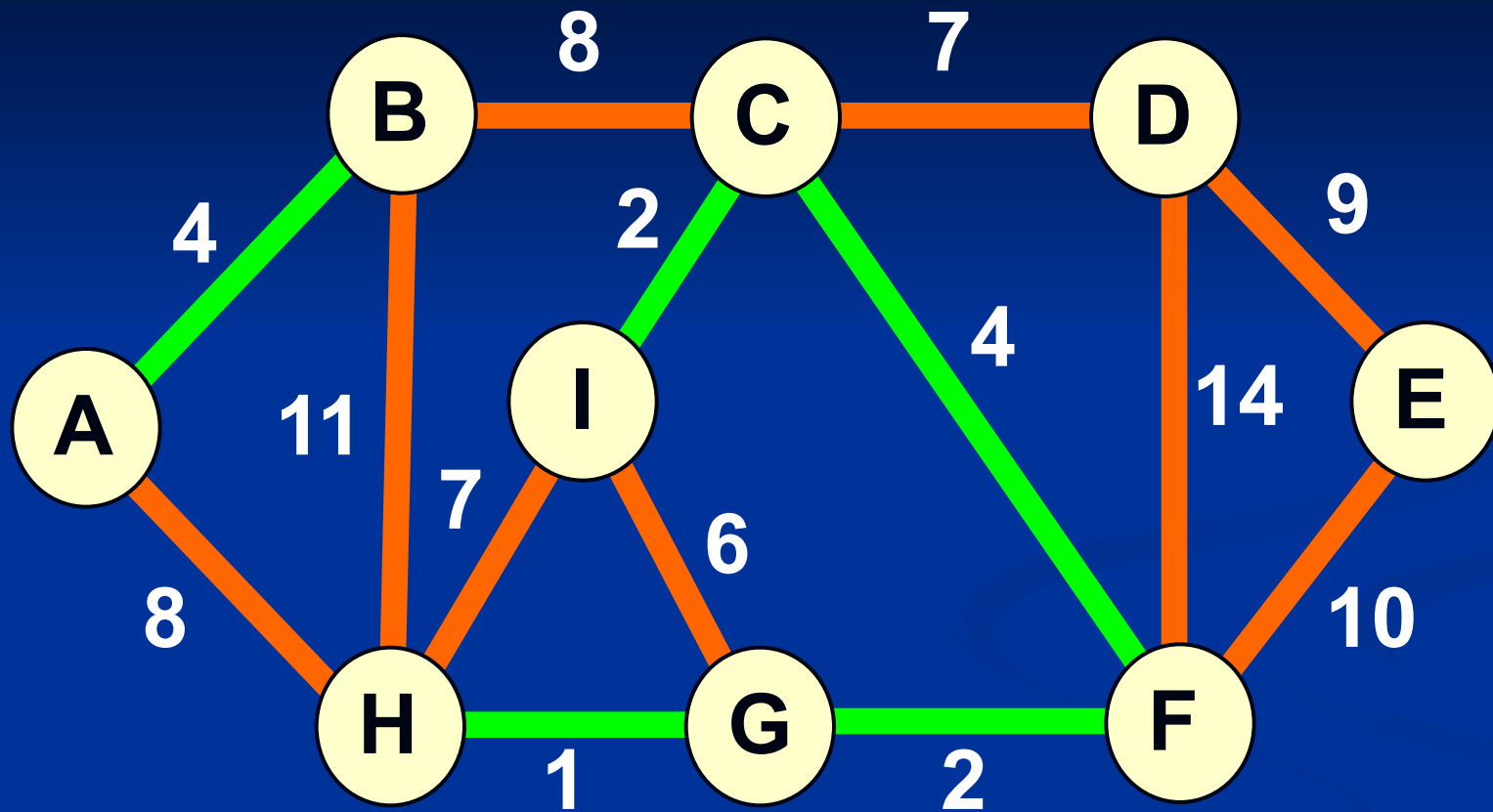
- Суммарная длина деревьев = 5

Алгоритм Краскала шаг 4



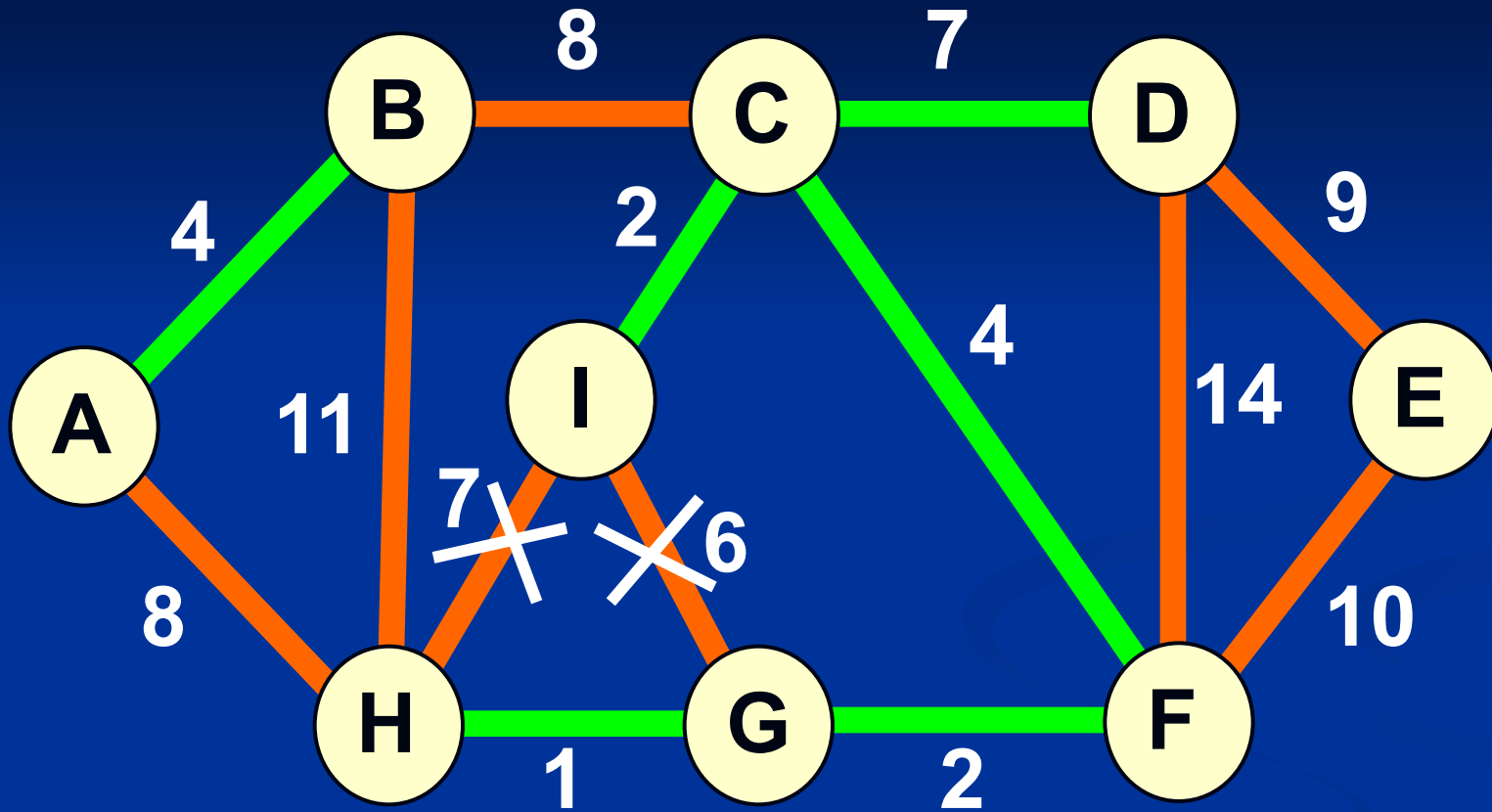
- Суммарная длина деревьев = 9

Алгоритм Краскала шаг 5



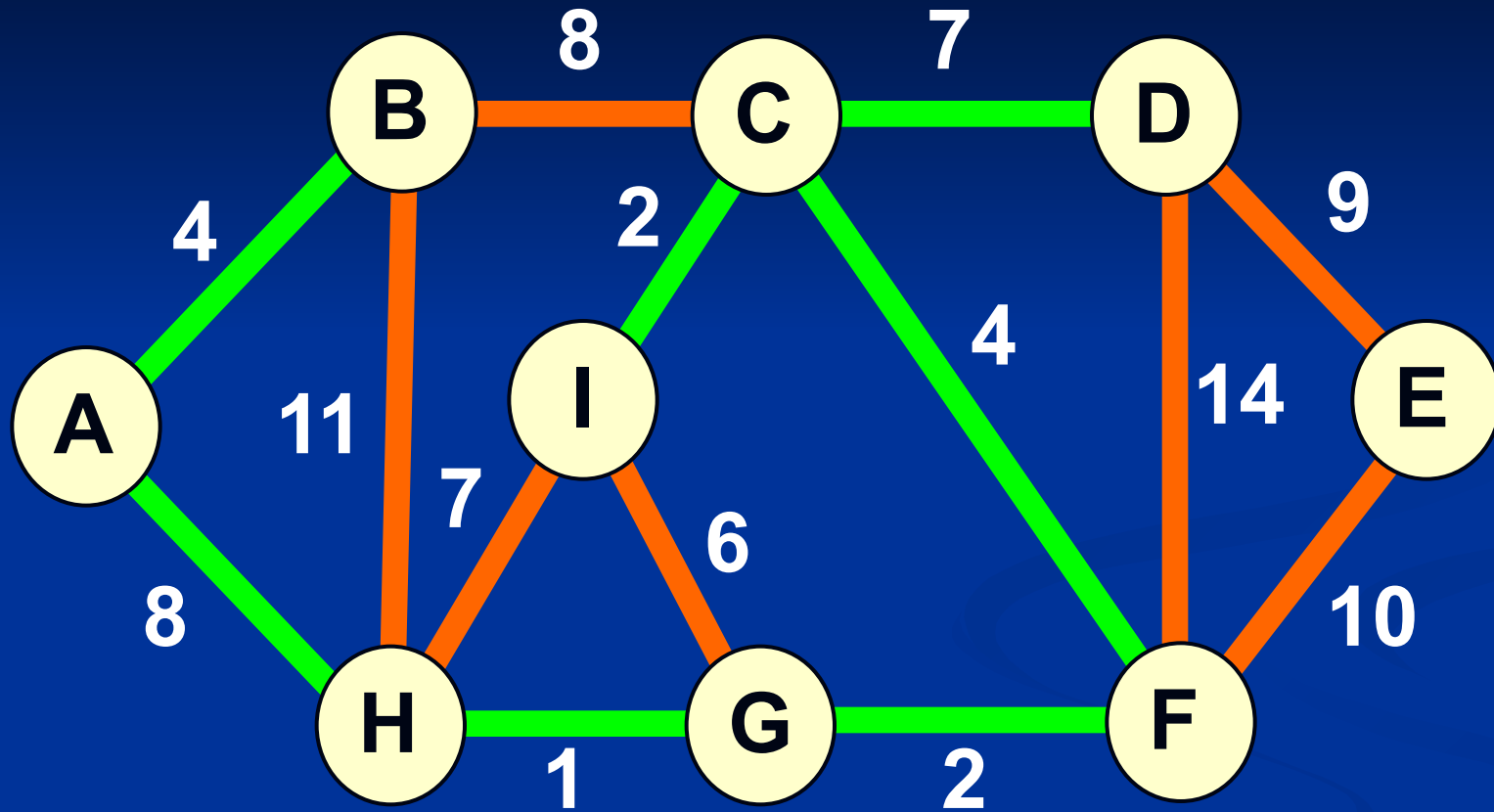
- Суммарная длина деревьев = 13

Алгоритм Краскала шаг 6



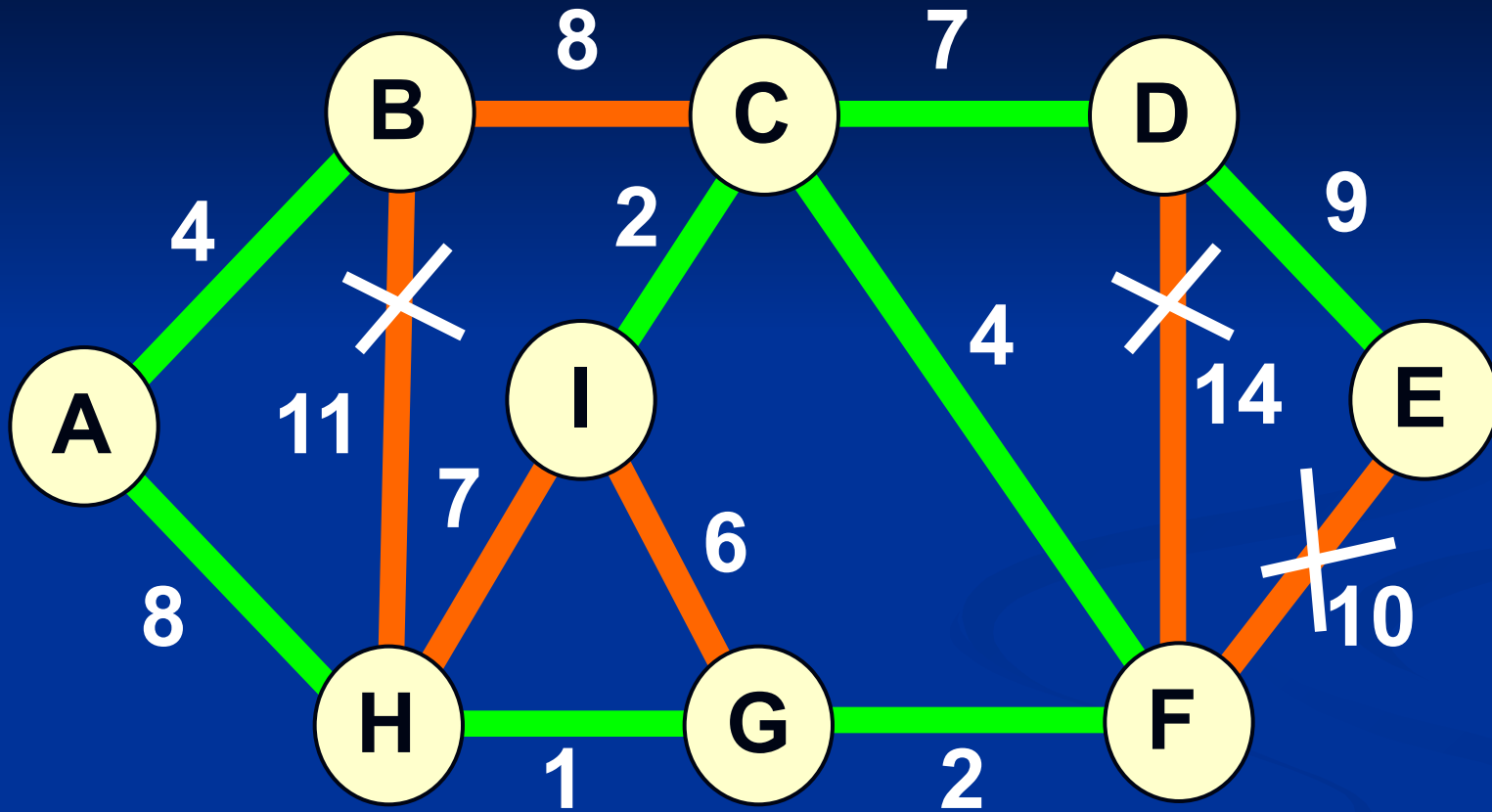
- Суммарная длина деревьев = 20

Алгоритм Краскала шаг 7



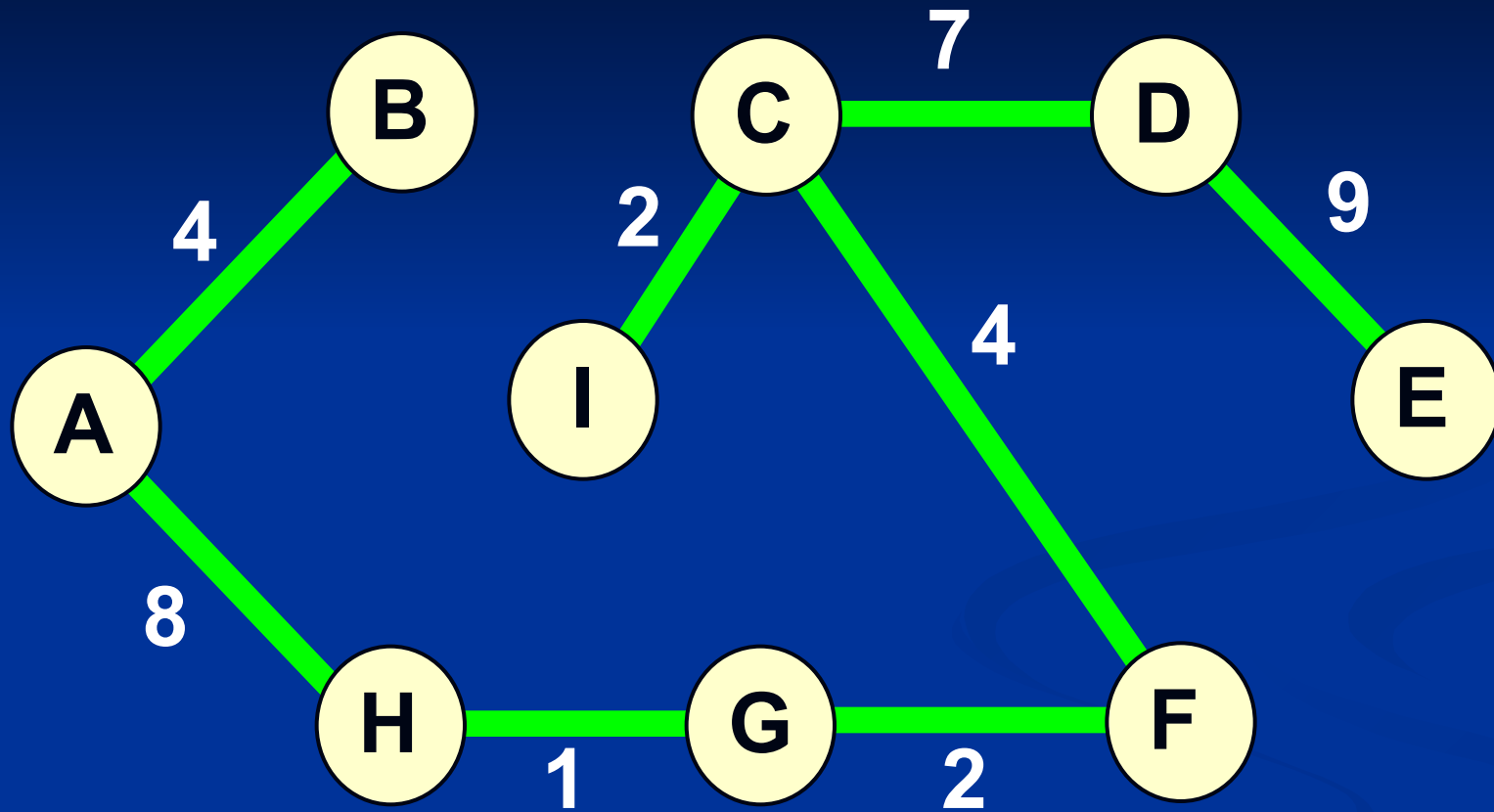
- Суммарная длина деревьев = 28

Алгоритм Краскала шаг 8



- Суммарная длина деревьев = 37

Алгоритм Краскала шаг 9

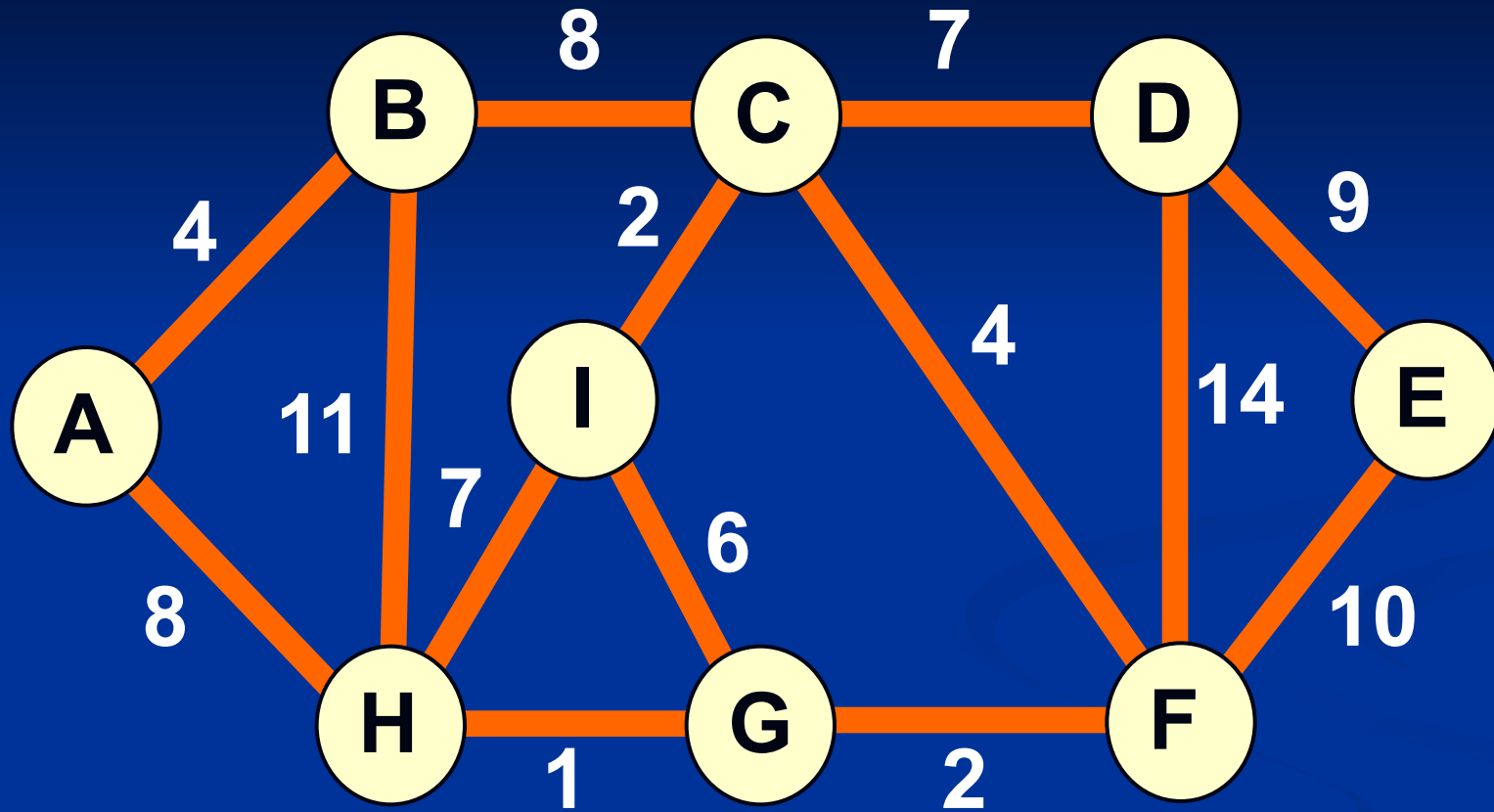


- Суммарная длина деревьев = 37

Алгоритм Прима

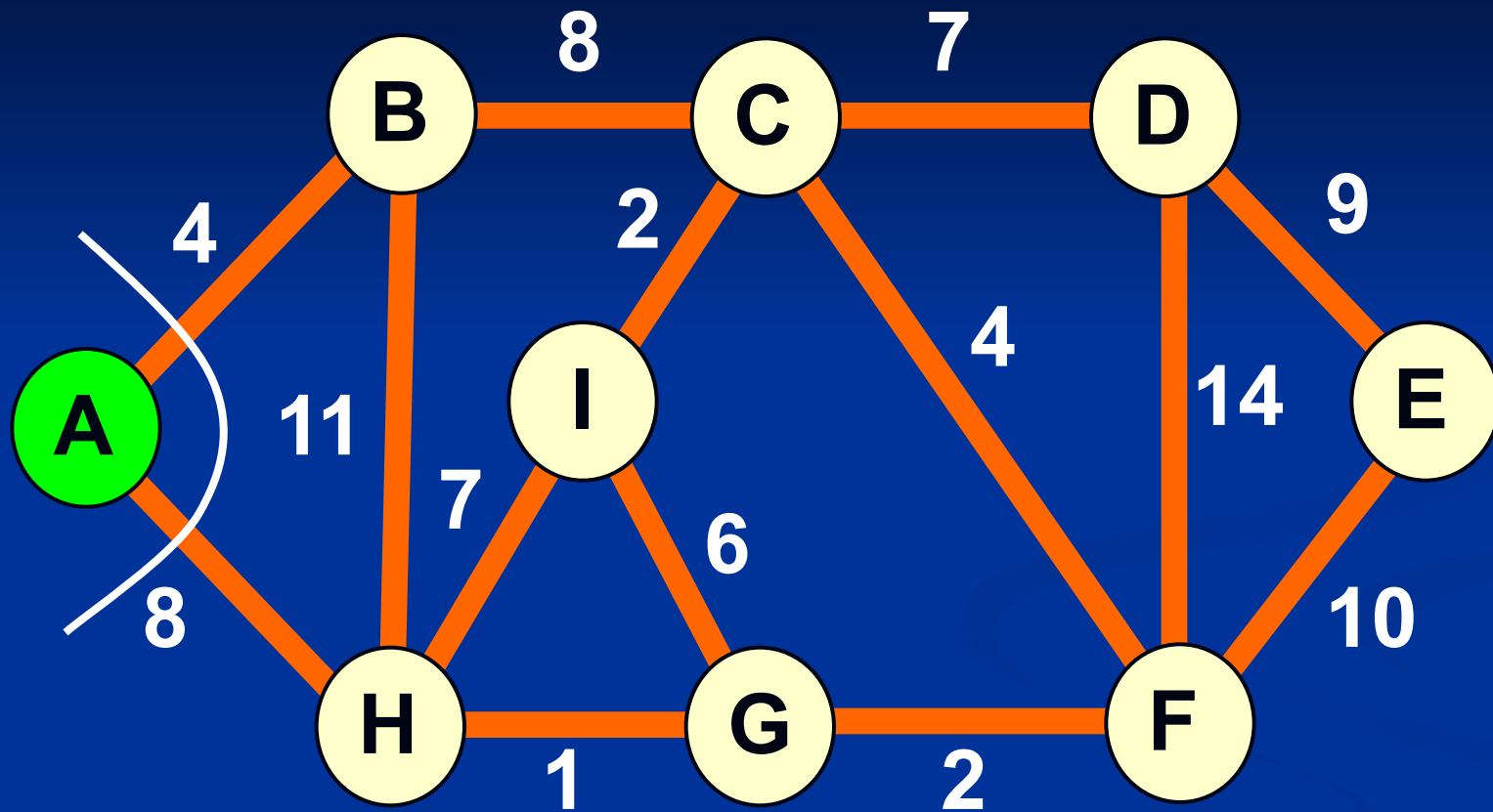
- Начало алгоритма: с произвольной вершины
- К текущему дереву присоединяется смежная вершина с кратчайшим ребром.
- Окончание алгоритма: либо все вершины подключены, либо невозможно подключить ни одно ребро.

Алгоритм Прима шаг 0



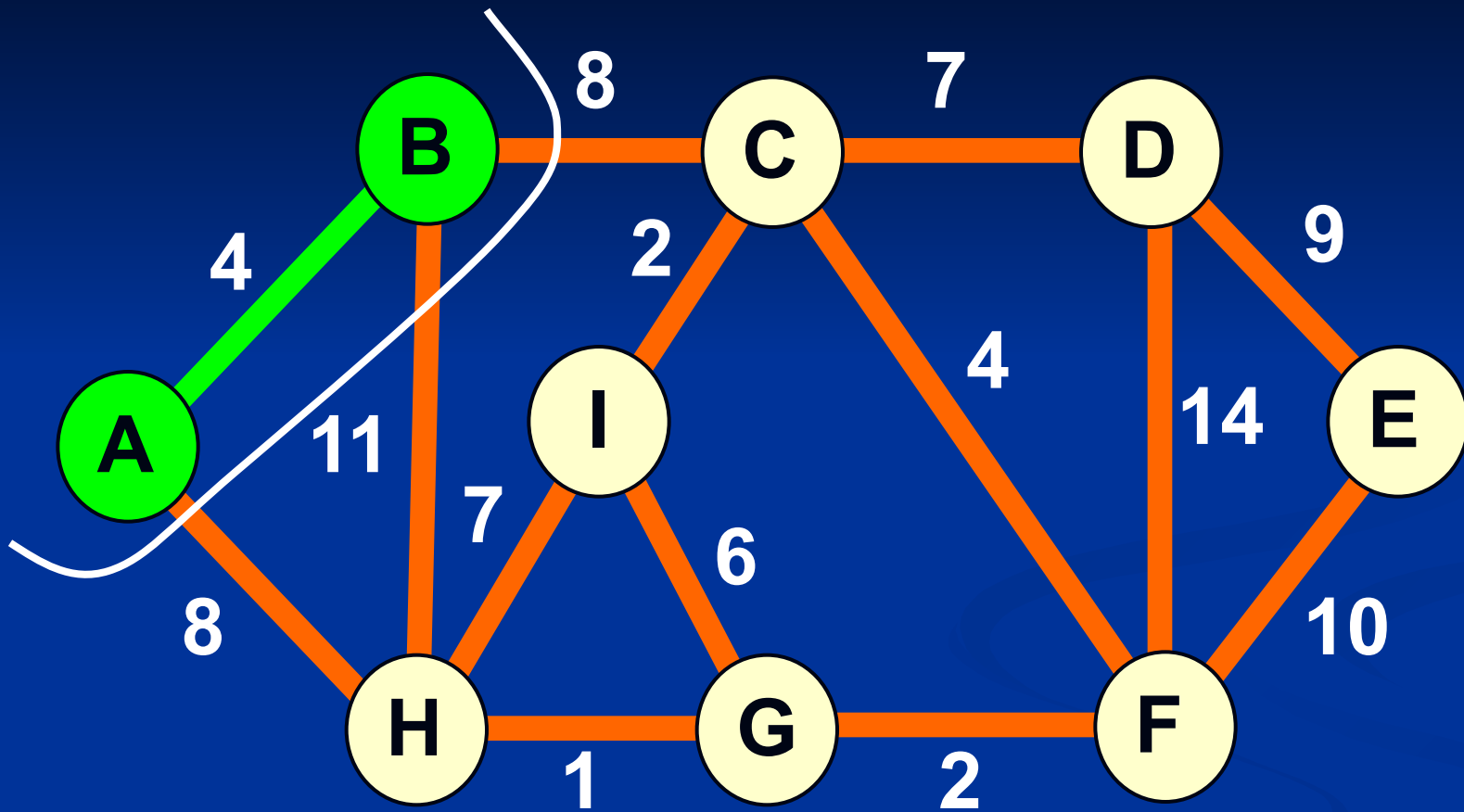
- Суммарная длина дерева = 0

Алгоритм Прима шаг 1



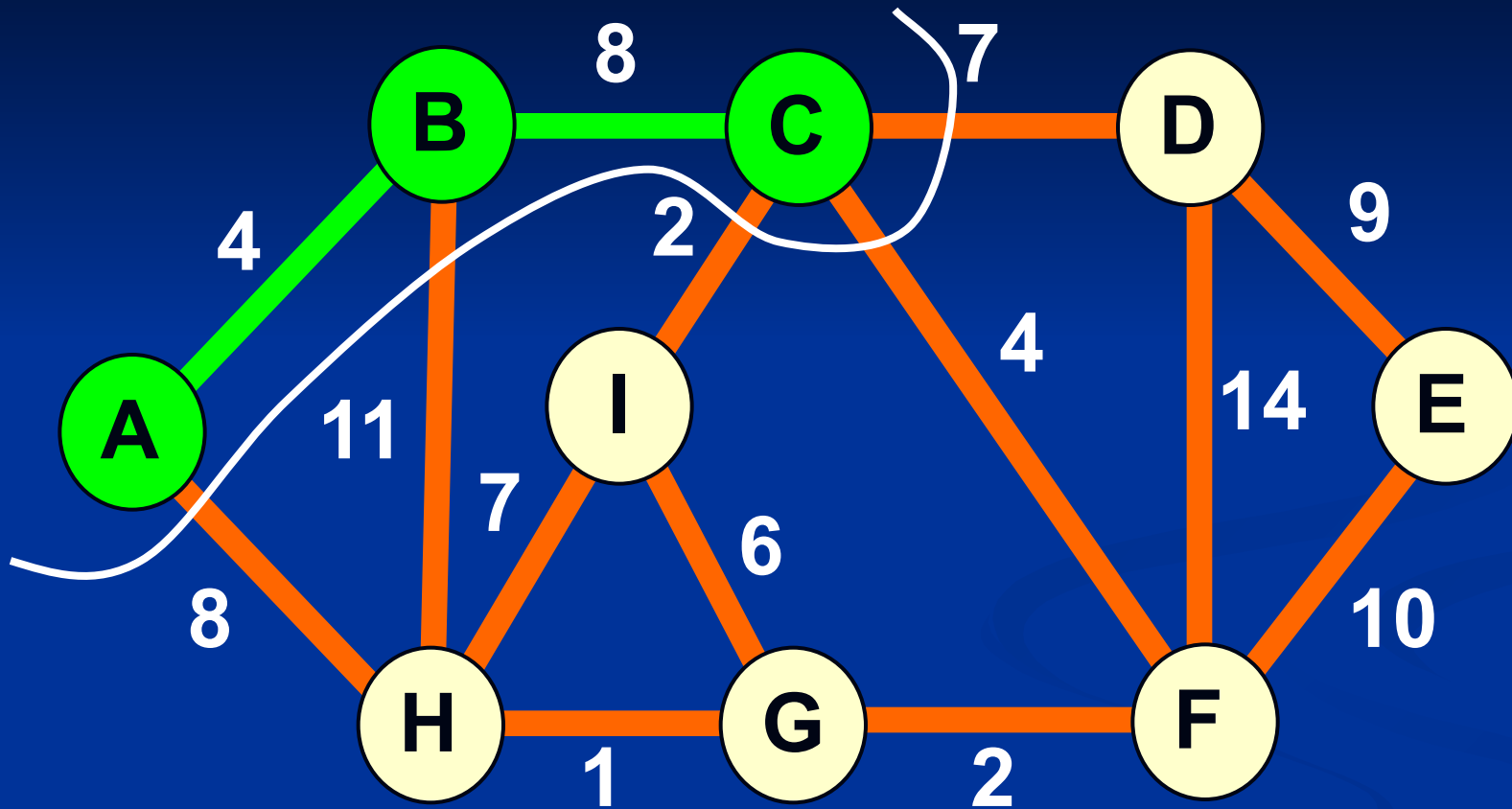
- Суммарная длина дерева = 0

Алгоритм Прима шаг 2



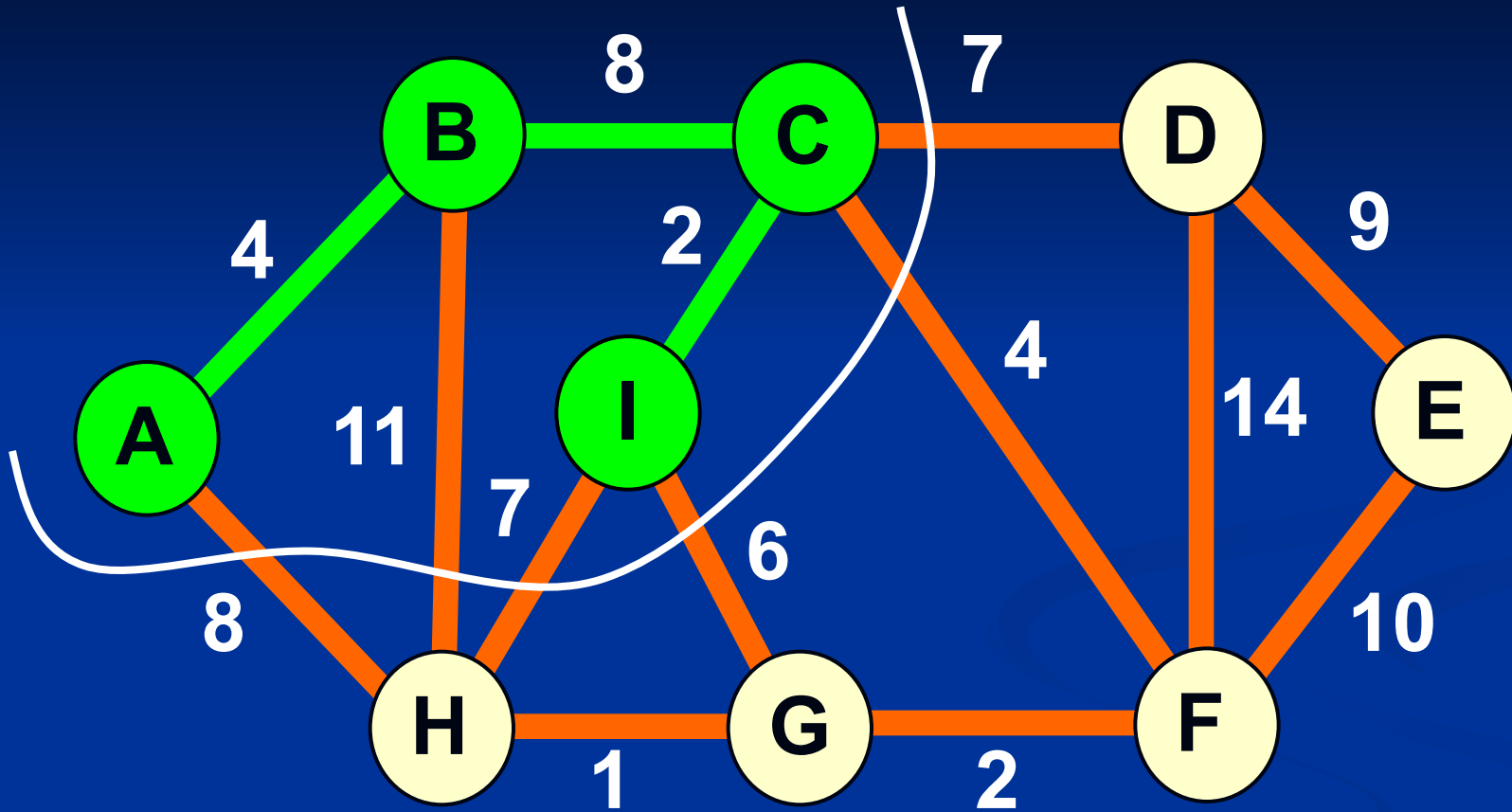
- Суммарная длина дерева = 4

Алгоритм Прима шаг 3



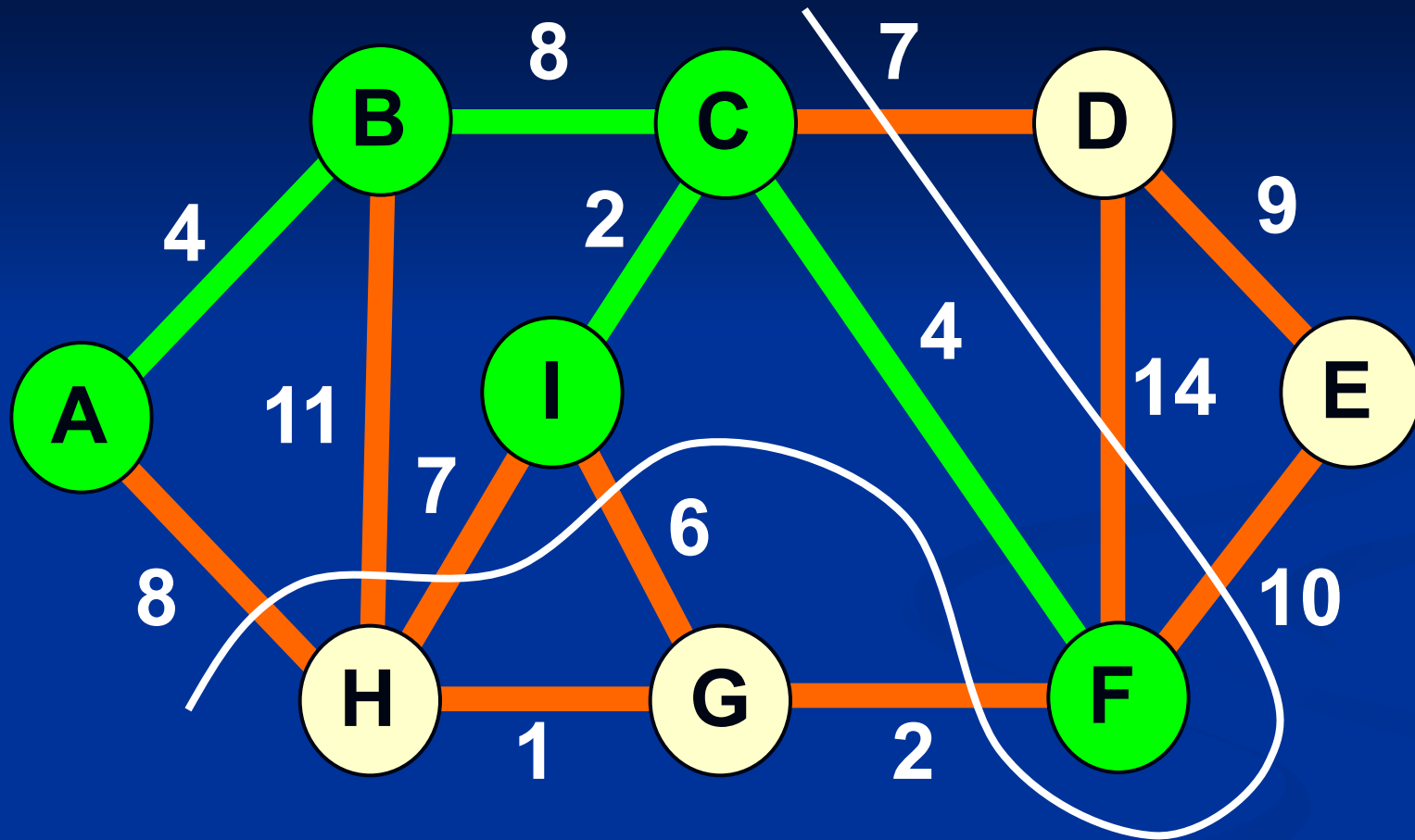
- Суммарная длина дерева = 12

Алгоритм Прима шаг 4



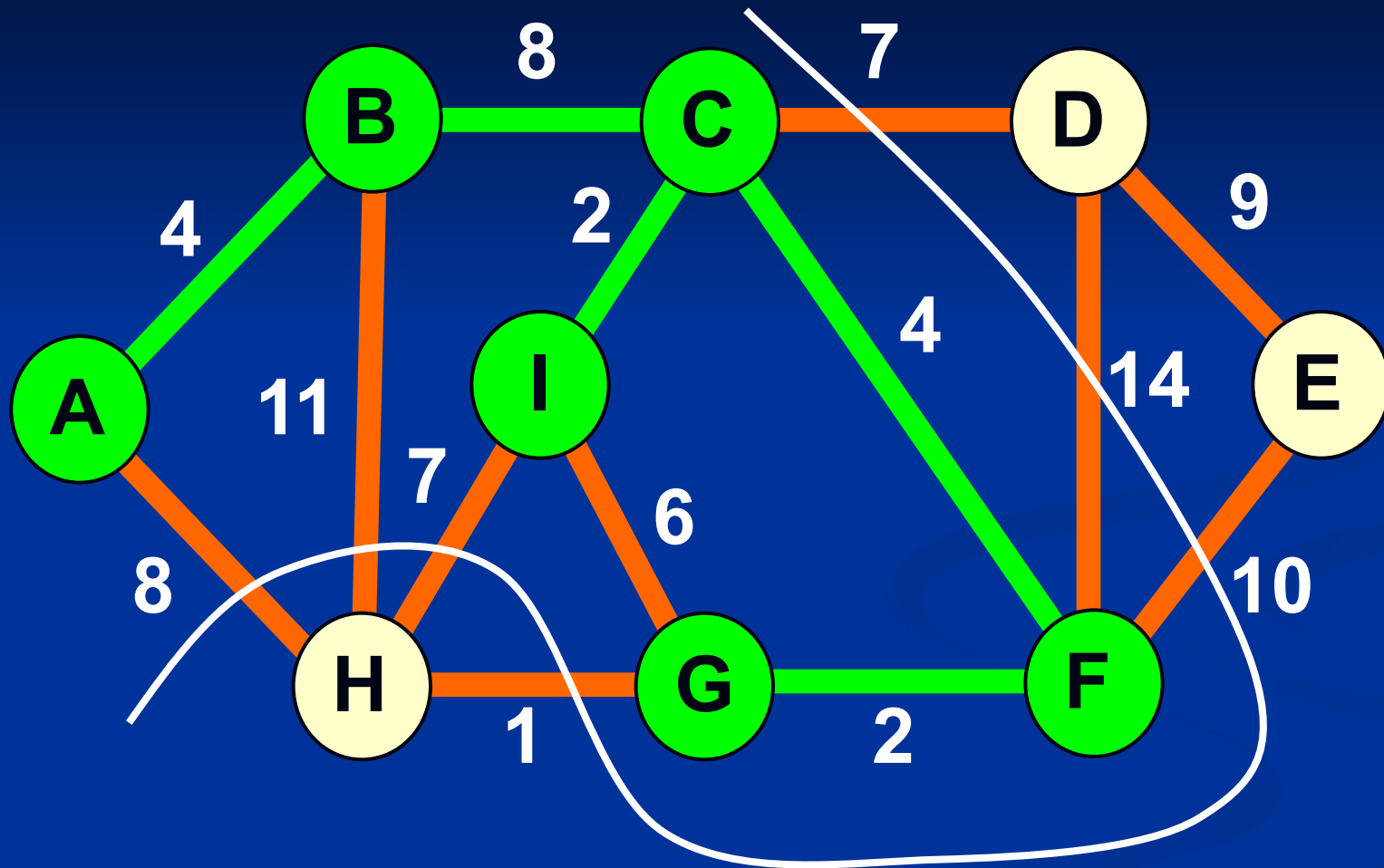
- Суммарная длина дерева = 14

Алгоритм Прима шаг 5



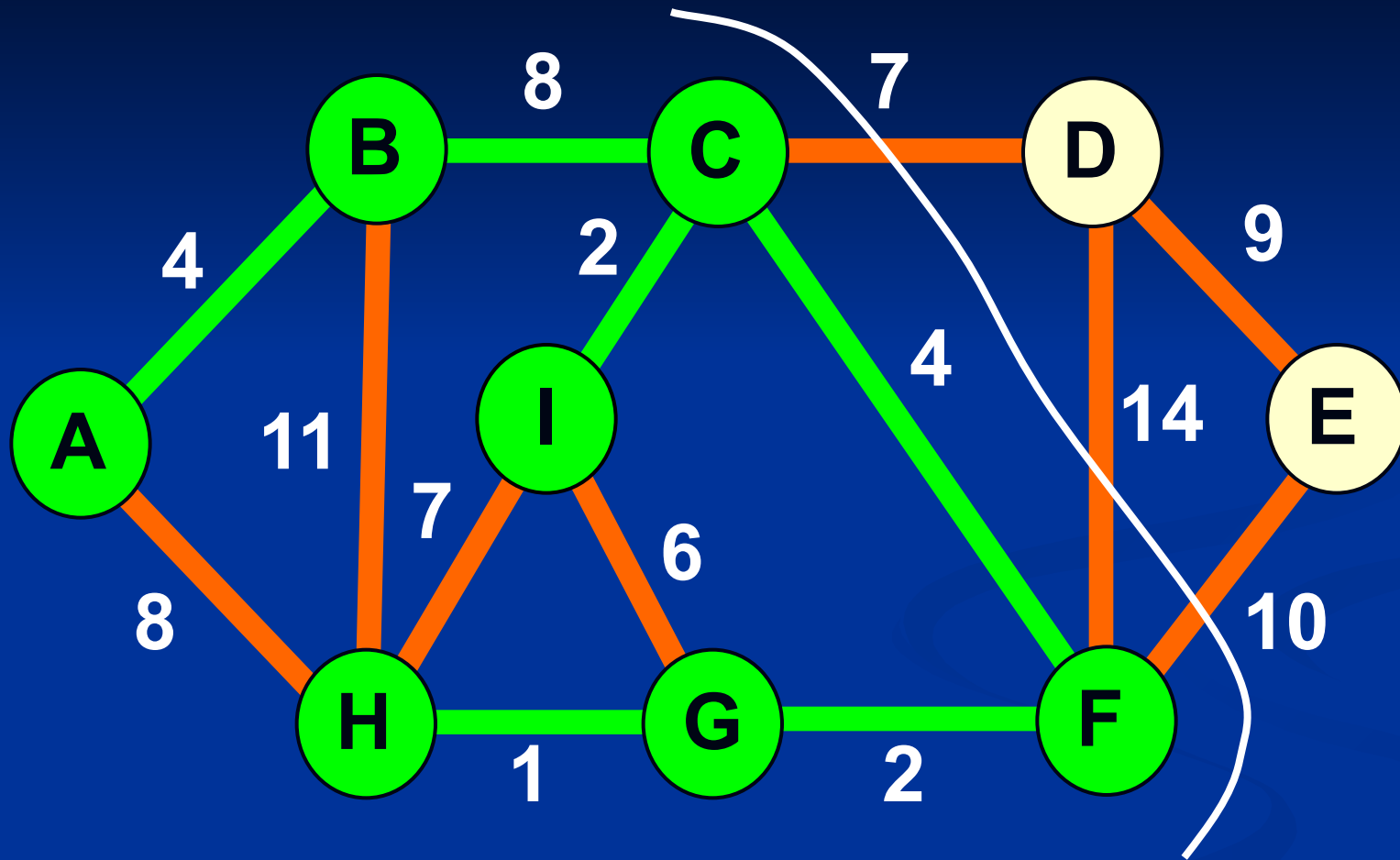
- Суммарная длина дерева = 18

Алгоритм Прима шаг 6



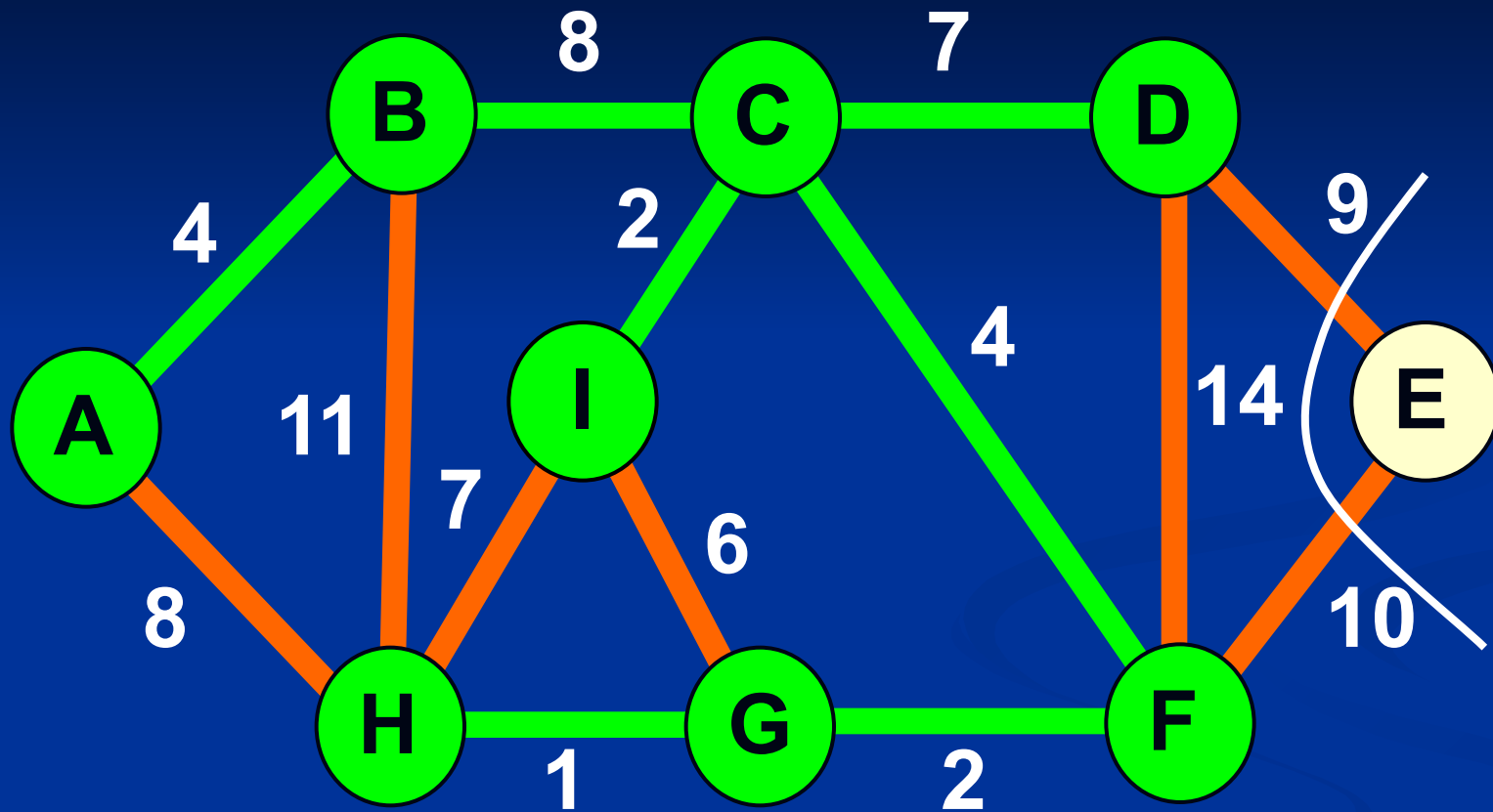
■ Суммарная длина дерева = 20

Алгоритм Прима шаг 7



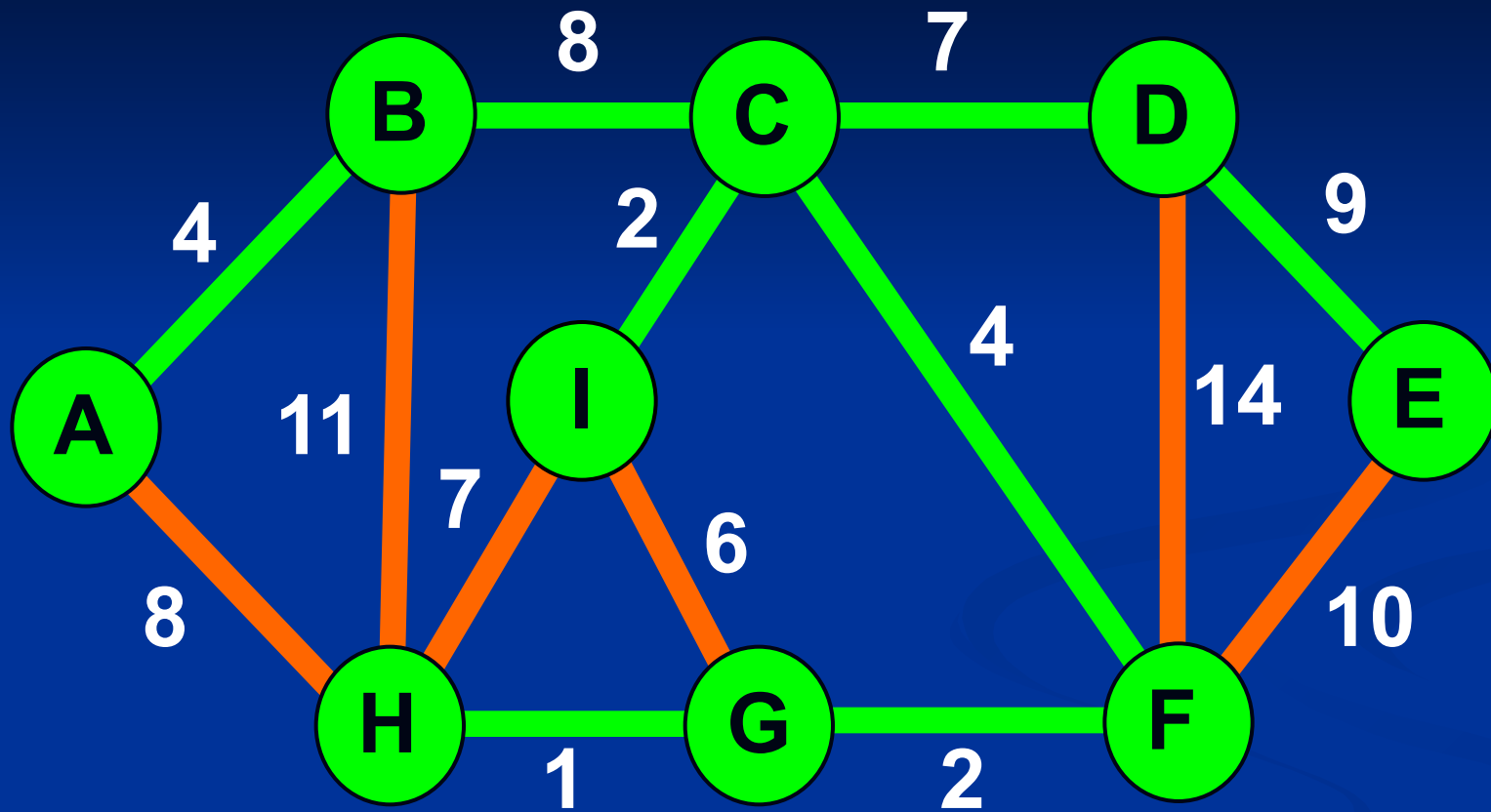
- Суммарная длина дерева = 21

Алгоритм Прима шаг 8



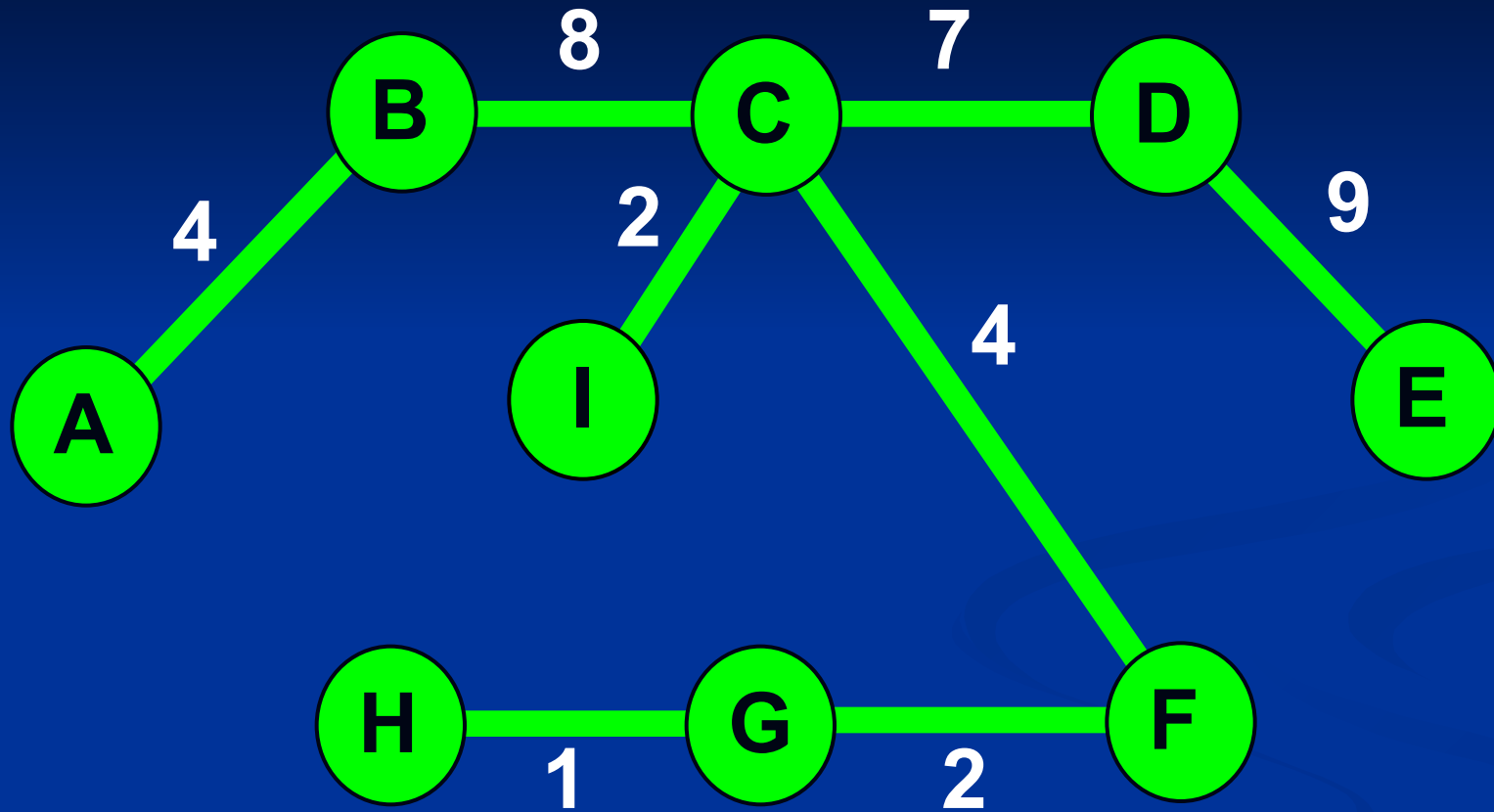
- Суммарная длина дерева = 28

Алгоритм Прима шаг 9



- Суммарная длина дерева = 37

Алгоритм Прима шаг 10

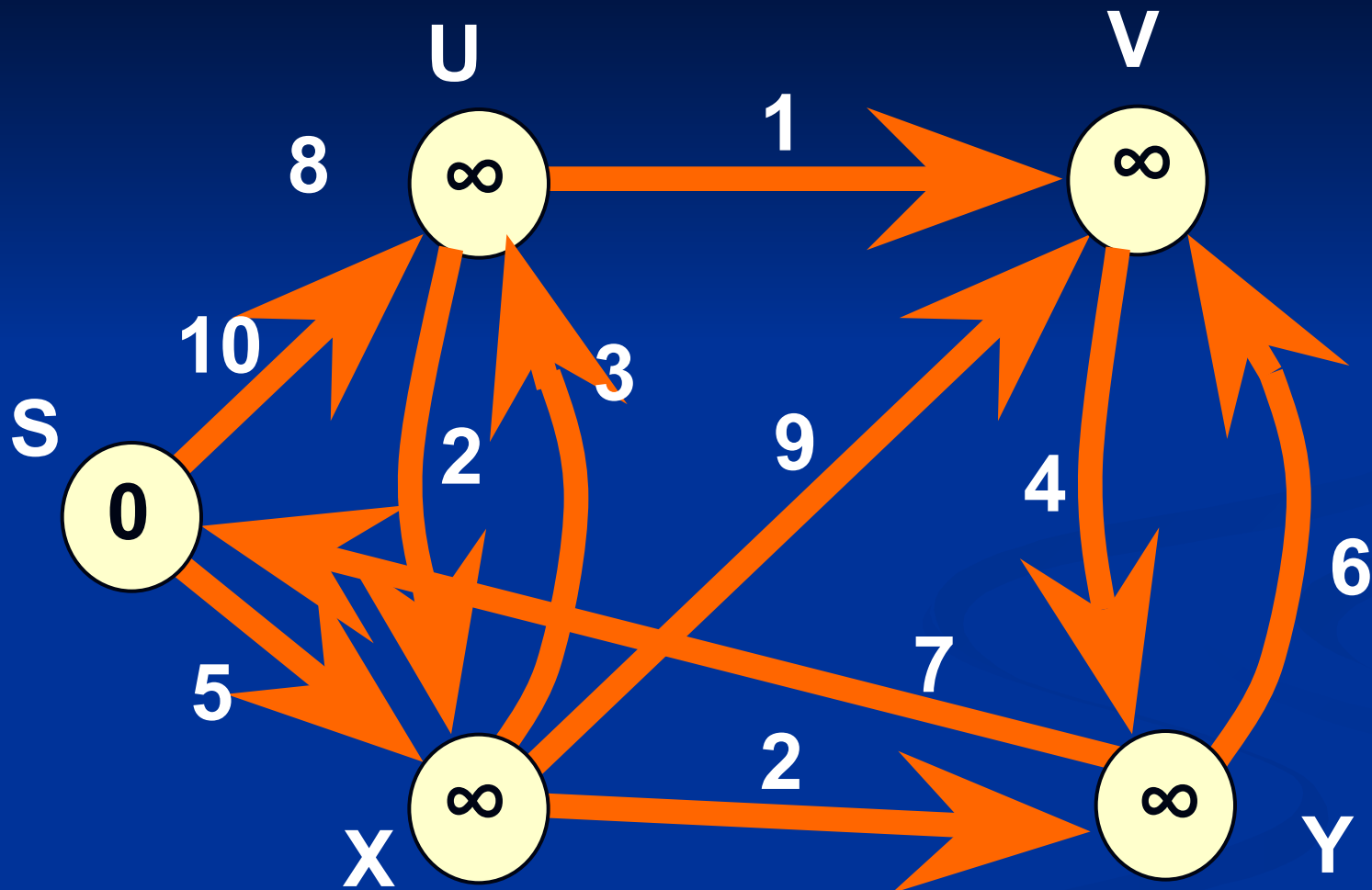


- Суммарная длина дерева = 37

КРАТЧАЙШИЕ ПУТИ В ГРАФЕ

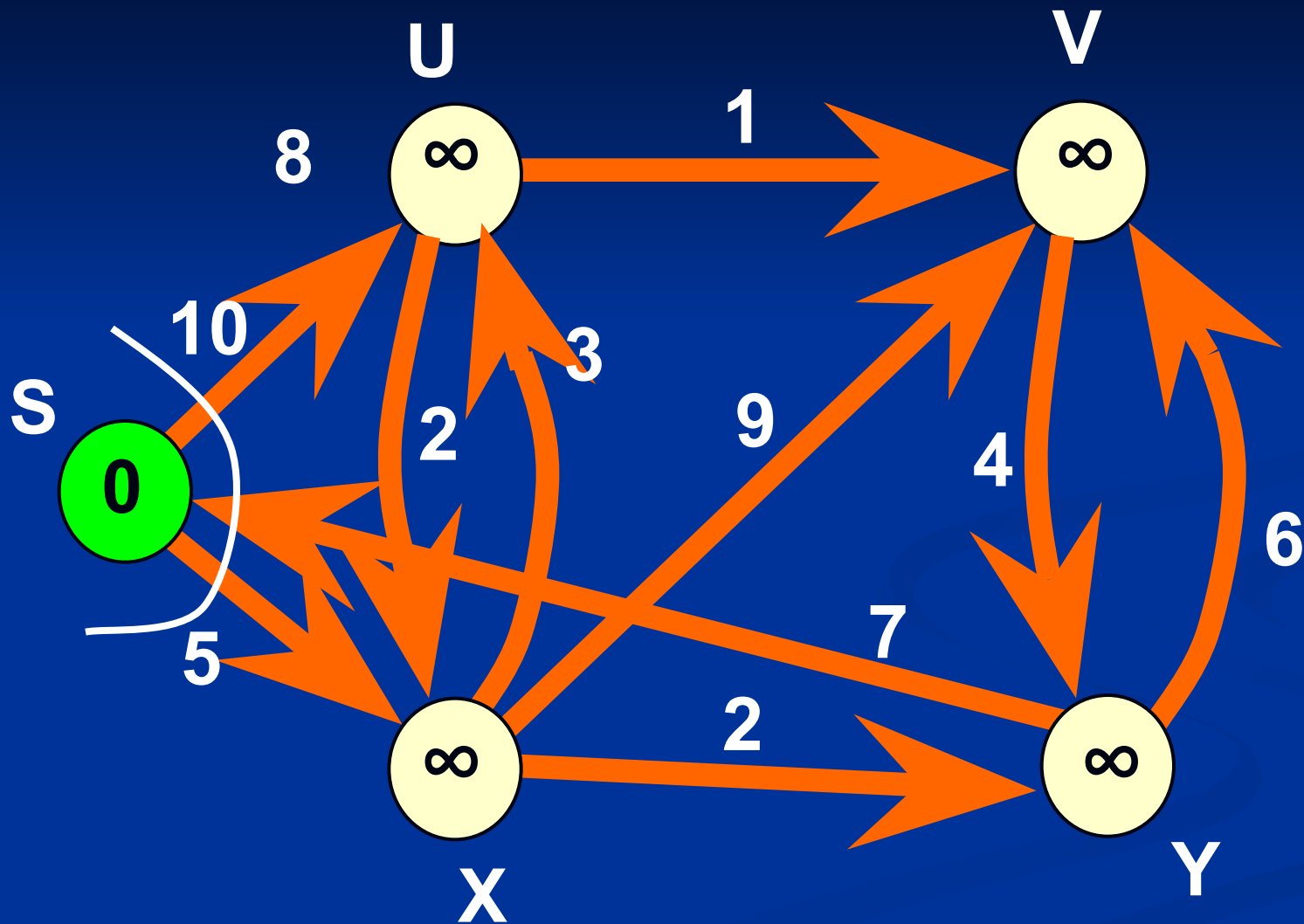
- Алгоритм Дейкстры (Дийкстры)
- Алгоритм Ли
- Алгоритм A^* (A-звездочка)

Алгоритм Дейкстры

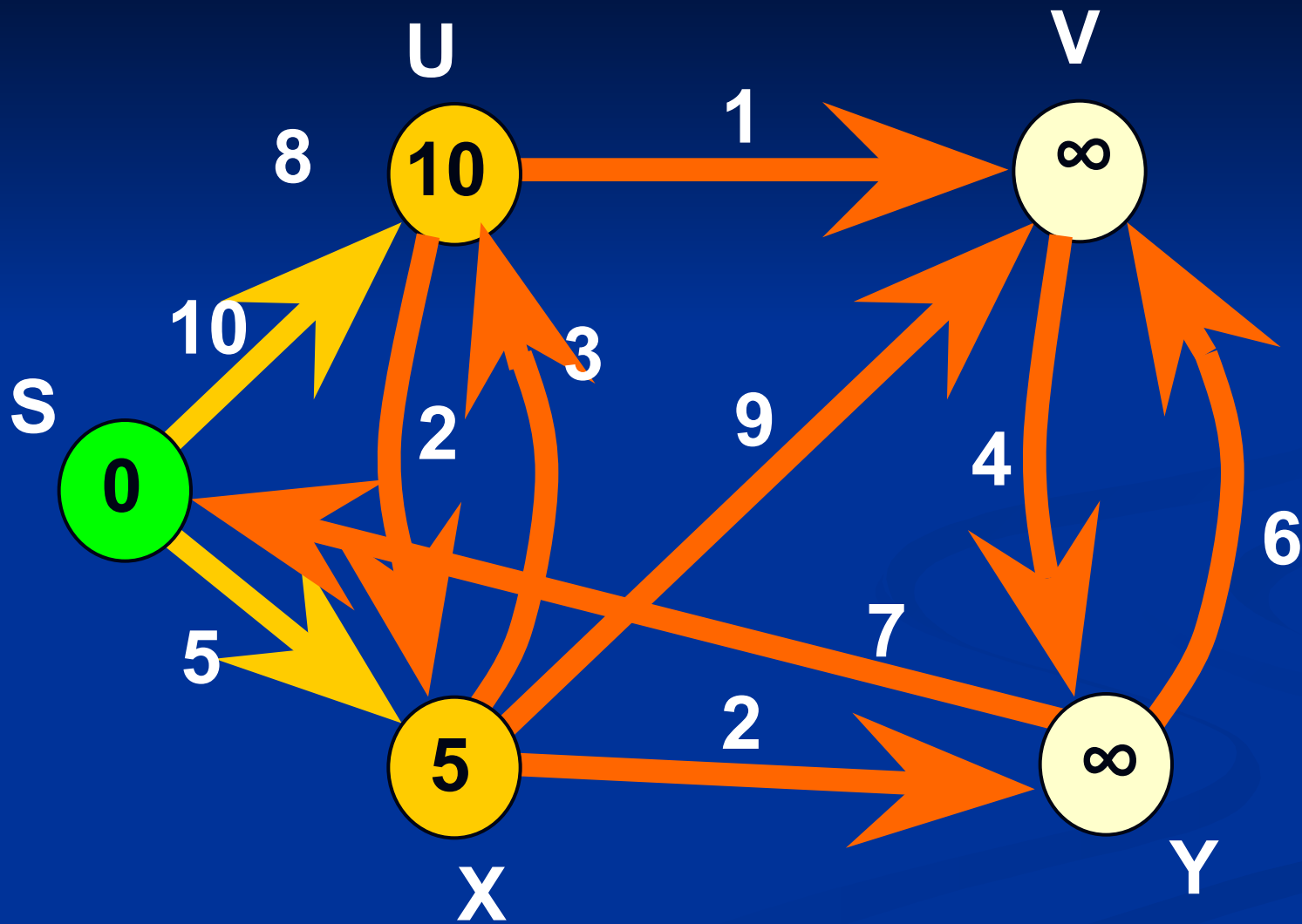


Ищем путь из S V

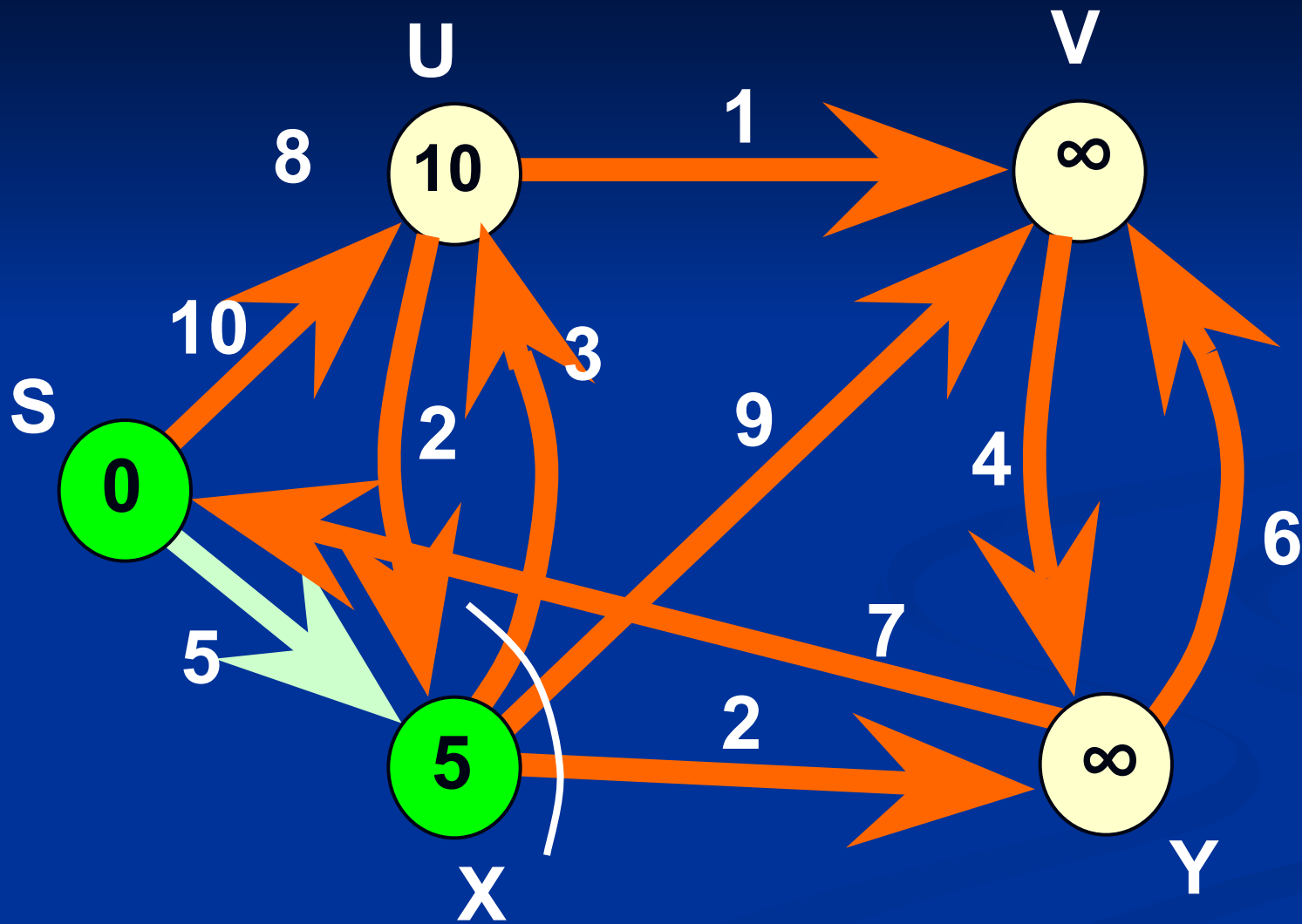
Алгоритм Дейкстры



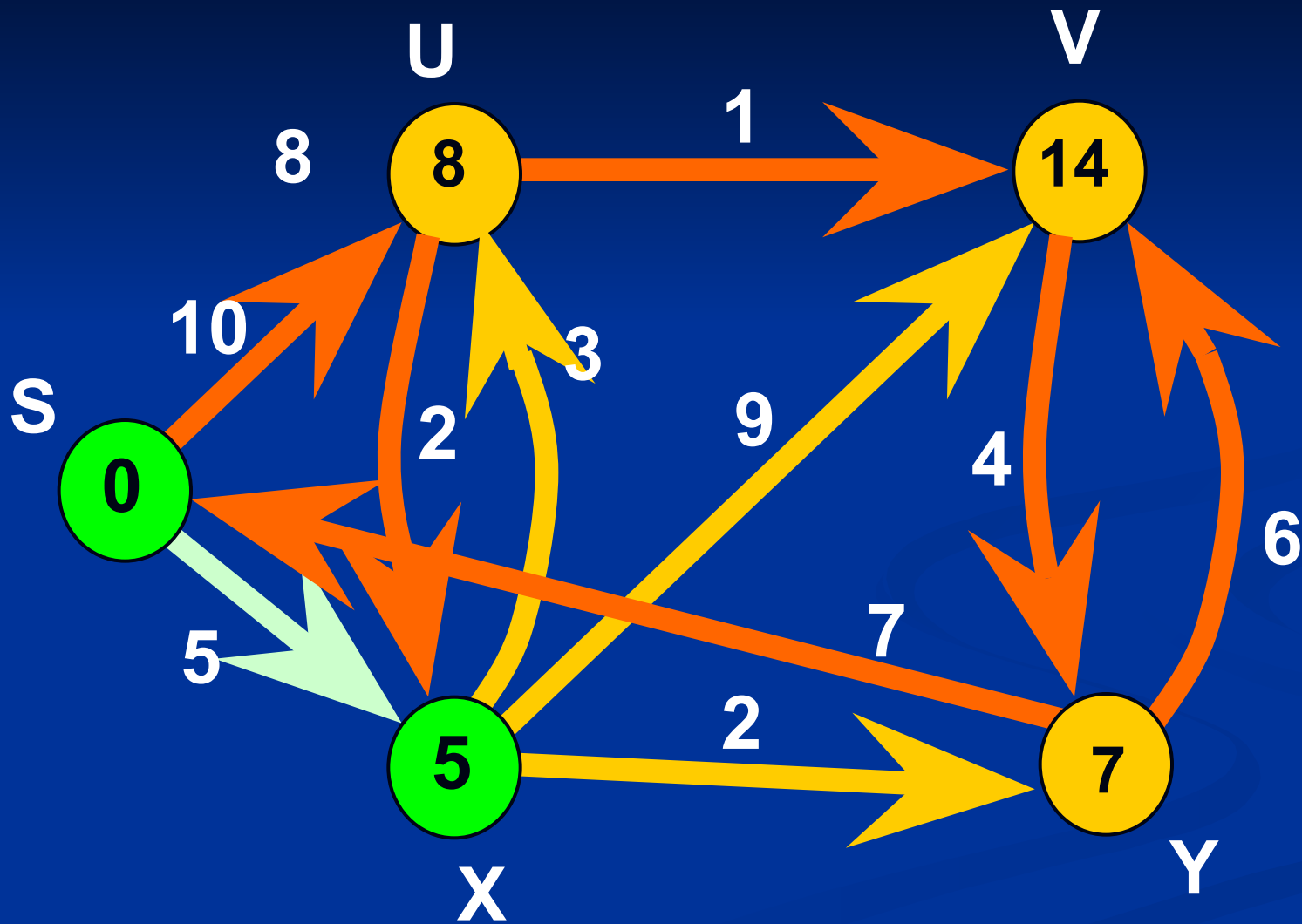
Алгоритм Дейкстры



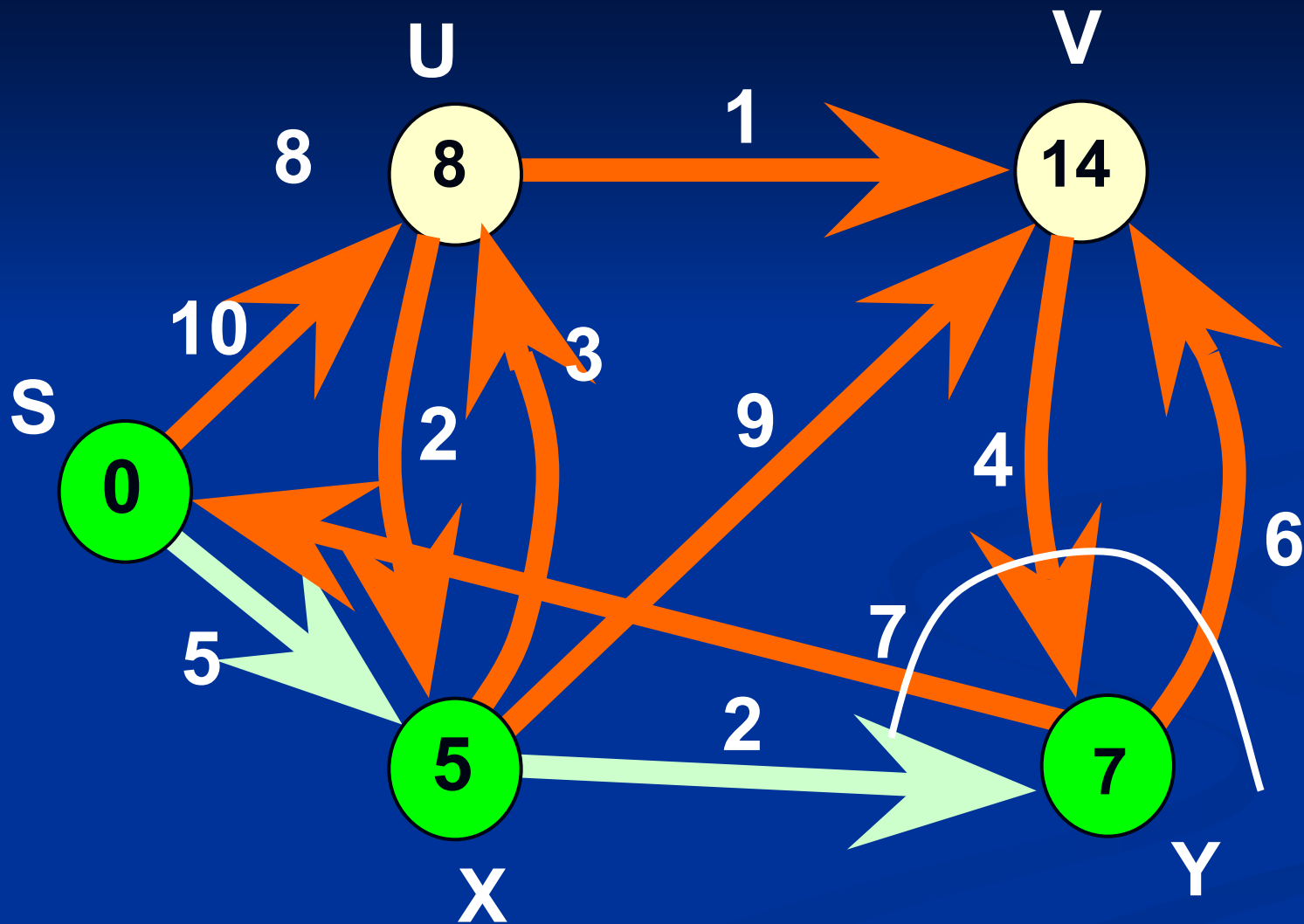
Алгоритм Дейкстры



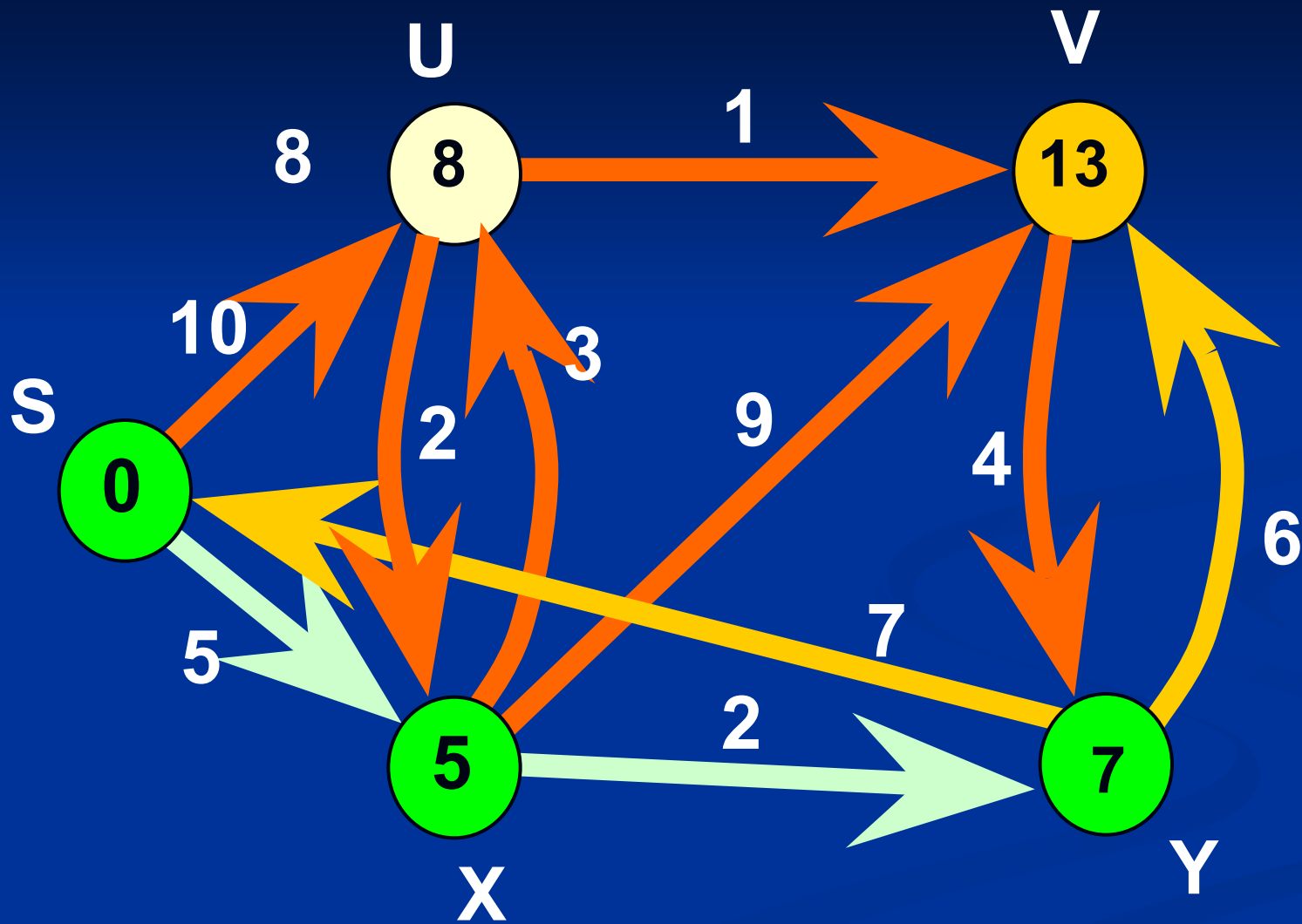
Алгоритм Дейкстры



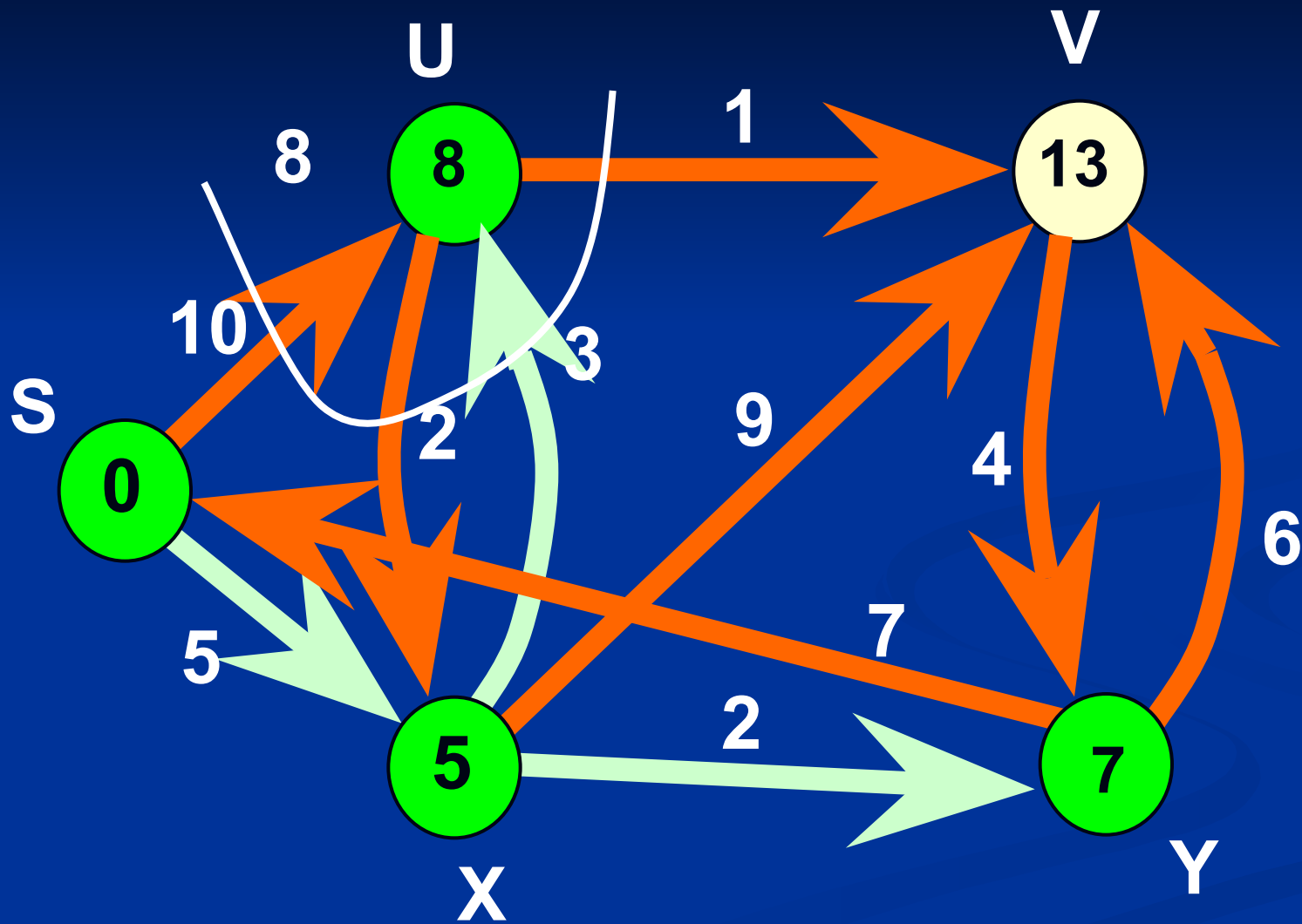
Алгоритм Дейкстры



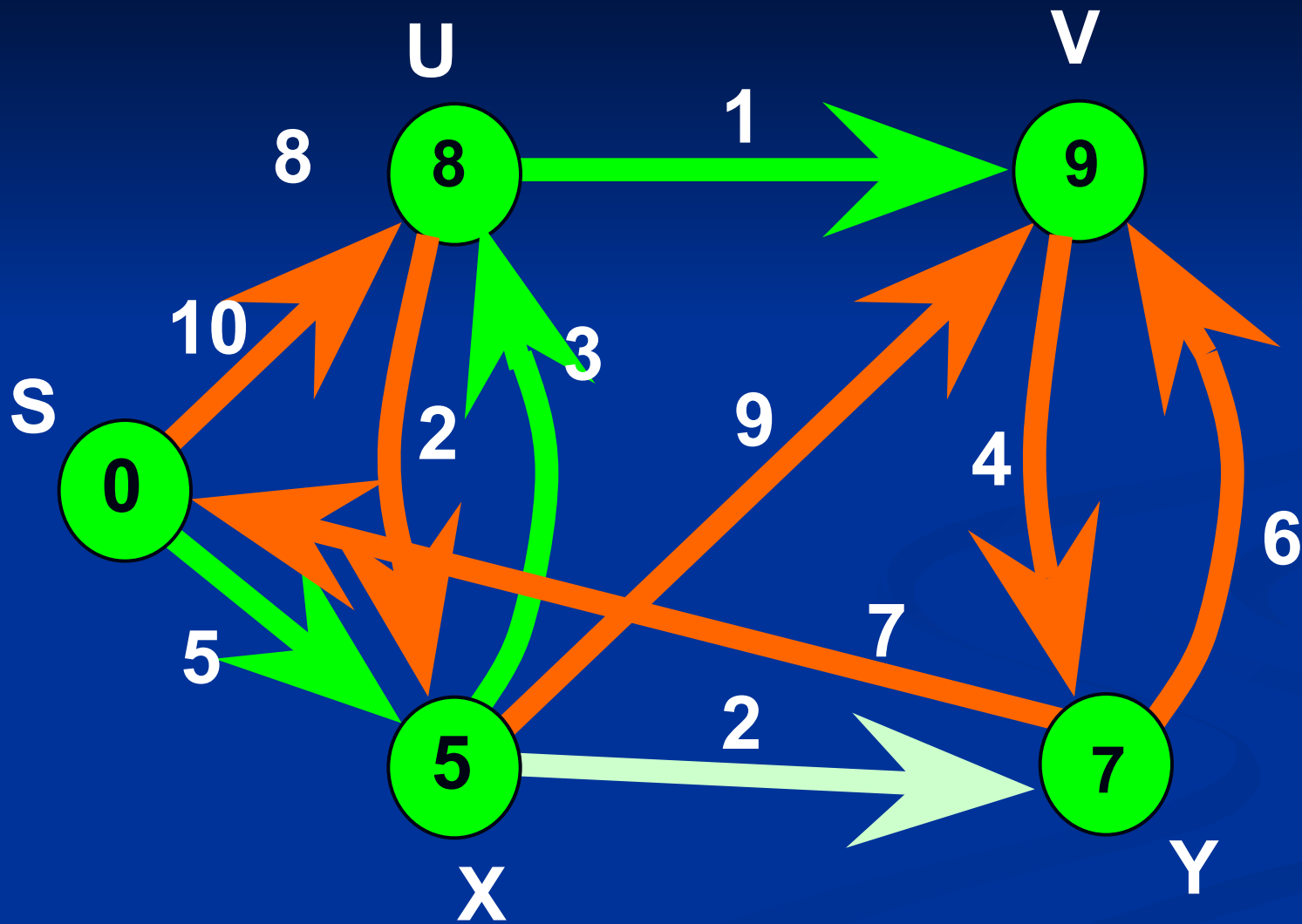
Алгоритм Дейкстры



Алгоритм Дейкстры

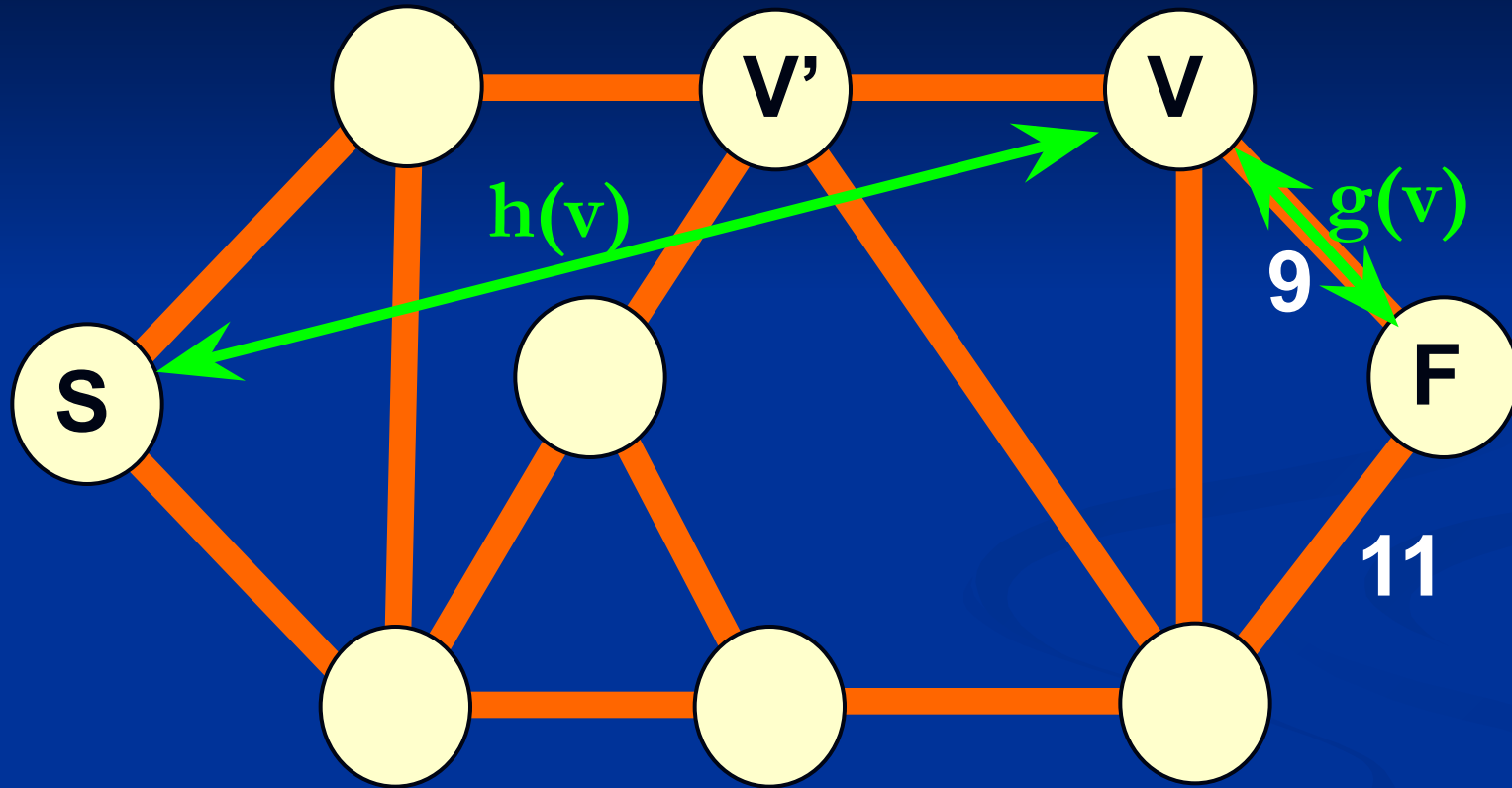


Алгоритм Дейкстры



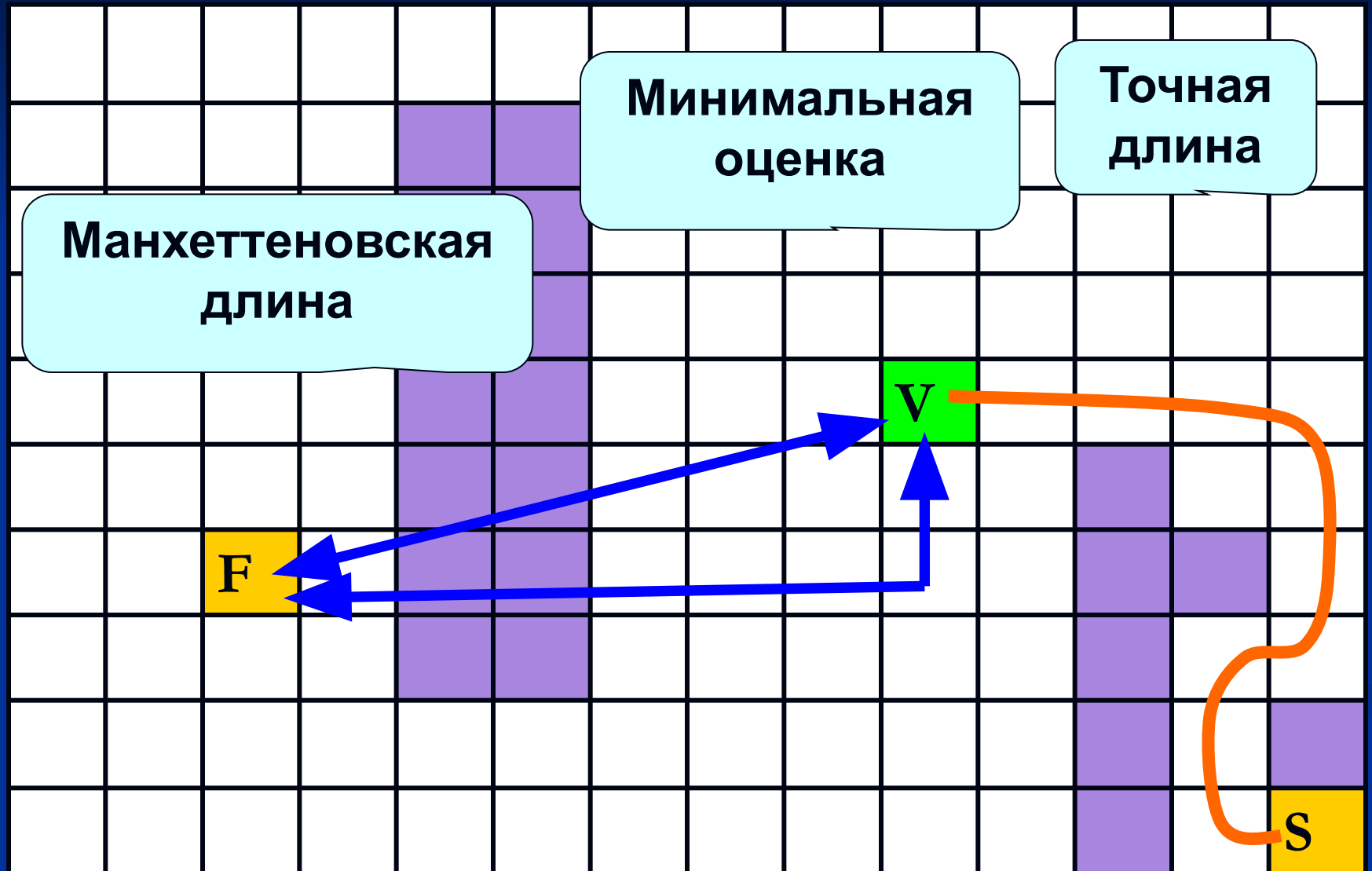
Алгоритм A*

(Алгоритм оптимального поиска)



$$F(v) = g(v) + h(v)$$

Оценка длины пути



Алгоритм A*

- $g(v)$ – стоимость пути от финиша до вершины v .
- $h(v)$ – нижняя оценка стоимости пути от вершины v до старта.
- $f(v) = g(v) + h(v)$ – нижняя оценка стоимости пути от старта к финишу через вершину v .

Алгоритм A*

1. Среди вершин, смежных с конечной найти вершину V , имеющую наименьшую оценку $f(v)$.
2. Если вершина V не смежна с начальной, то среди вершин, достижимых из V найти вершину V' с наименьшим значением $f(v)$.
3. Продолжать, пока не будет достигнута начальная вершина.