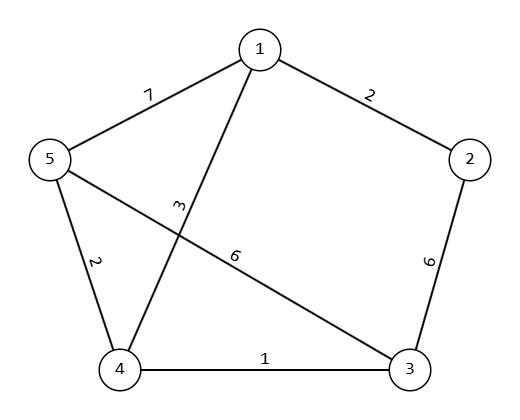
**Задание 1**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 |
| 1 |  | 2 |  | 3 | 7 |
| 2 | 2 |  | 6 |  |  |
| 3 |  | 6 |  | 1 | 6 |
| 4 | 3 |  | 1 |  | 2 |
| 5 | 7 |  | 6 | 2 |  |



Связность

Представим граф в виде списка смежности

1: [2,4,5]

2: [1,3]

3: [2,4,5]

4: [3,5,1]

5: [4,3,1]

Переходим к алгоритму

1. Начинаем с вершины 1:

* Посещаем 1, добавляем в компоненту: {1}
* Переходим к 2:
* Посещаем 2, добавляем в компоненту: {1,2}
* Переходим к 3:
* Посещаем 3, добавляем в компоненту: {1,2,3}
* Переходим к 4:
* Посещаем 4, добавляем в компоненту: {1,2,3,4}
* Переходим к 5:
* Посещаем 5, добавляем в компоненту: {1,2,3,4,5}
* Все соседние вершины посещены, завершаем поиск в глубину для вершины 1.

1. Проверяем остальные вершины (2,3,4,5), но они уже посещены.

Граф состоит из одной компоненты связности: {1,2,3,4,5}

Алгоритм Прима

1. Ищем не посещённую вершину с наименьшей меткой (1). Проверяем все вершины смежные с 1 вершиной. Обновляем метки: 2:2<*,* 4: 3<, 5: 7<. Ищем не посещённую вершину с наименьшей меткой. Добавляем ребро (1,2).
2. Проверяем все смежные с 2 вершины и обновляем метку: 3: 6<. Ищем вершину с наименьшей меткой. Добавляем ребро (1,4).
3. Проверяем все смежные с 4 вершины и обновляем метку: 3: 1<6, 5: 2<7. Ищем не посещённую вершину с наименьшей меткой. Добавляем ребро (3,4).
4. Проверяем все смежные с 3 вершины и ищем не посещённую вершину с наименьшей меткой. Добавляем ребро (4,5).

Итоговое дерево T = {(1,2), (1,4), (3,4), (4,5)}

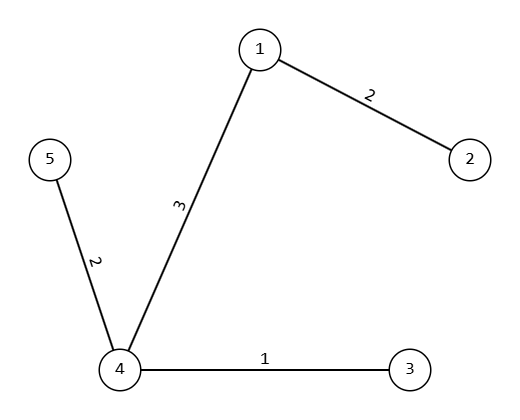
Вес минимального остовного дерева: 8.

Алгоритм Крускала

|  |  |  |  |
| --- | --- | --- | --- |
| Ребра | Вес | Добавляем ребро или нет | Остовное дерево |
| (3,4) | 1 | + | Т1 = {(3,4)} |
| (1,2) | 2 | + | Т2 = {(3,4), (1,2)} |
| (4,5) | 2 | + | T3 = {(3,4), (1,2), (4,5)} |
| (1,4) | 3 | + | T4 = {(3,4), (1,2), (4,5), (1,4)} |
| (2,3) | 6 | - | T5 = T4 |
| (3,5) | 6 | - | T6 = T5 |
| (1,5) | 7 | - | T7 = T6 |

Итоговое дерево T = {(1,2), (1,4), (3,4), (4,5)}

Вес минимального остовного дерева: 8.

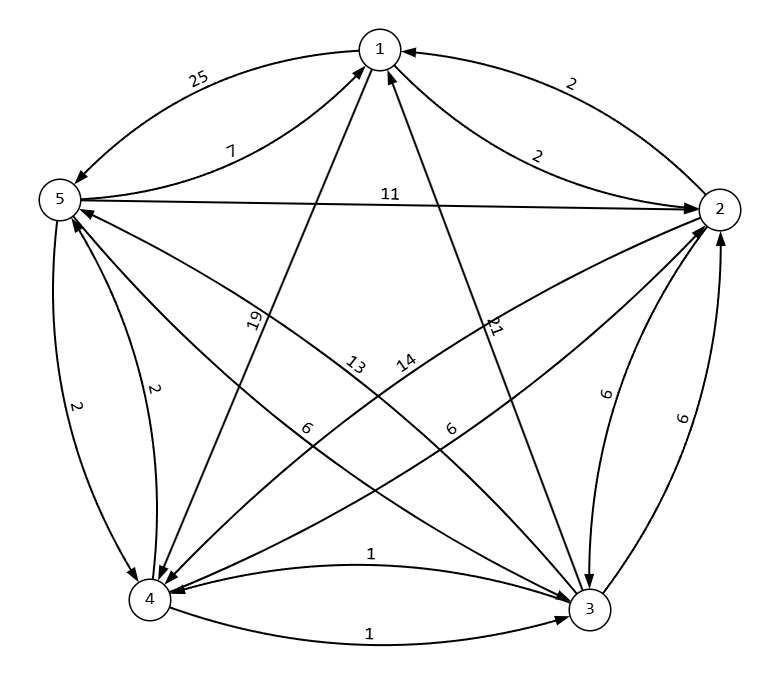


Мосты

|  |  |  |  |
| --- | --- | --- | --- |
| (1,2) | (1,4) | (3,4) | (4,5) |
| Не мост | Не мост | Не мост | Не мост |

**Задание 2**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **1** |  | 2 |  | 19 | 25 |
| **2** | 2 |  | 6 | 14 |  |
| **3** | 21 | 6 |  | 1 | 13 |
| **4** |  | 6 | 1 |  | 2 |
| **5** | 7 | 11 | 6 | 2 |  |



Алгоритм Дейкстры

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | Р |
| 0 | 0/1 | /6 | /6 | /6 | /6 | 1 |
| 1 |  | 2/1 |  | 19/1 | 25/1 | 2 |
| 2 |  |  | 8/2 | 16/2 | 25/1 | 3 |
| 3 |  |  |  | 9/3 | 21/3 | 4 |
| 4 |  |  |  |  | 11/4 | 5 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | Р |
| 0 | /6 | 0/2 | /6 | /6 | /6 | 2 |
| 1 | 2/2 |  | 6/2 | 14/2 |  | 1 |
| 2 |  |  | 6/2 | 14/2 | 27/1 | 3 |
| 3 |  |  |  | 7/3 | 19/3 | 4 |
| 4 |  |  |  |  | 9/4 | 5 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | Р |
| 0 | /6 | /6 | 0/3 | /6 | /6 | 3 |
| 1 | 21/3 | 6/3 |  | 1/3 | 13/3 | 4 |
| 2 | 21/3 | 6/3 |  |  | 3/4 | 5 |
| 3 | 10/5 | 6/3 |  |  |  | 2 |
| 4 | 8/2 |  |  |  |  | 1 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | Р |
| 0 | /6 | /6 | /6 | 0/4 | /6 | 4 |
| 1 |  | 6/4 | 1/4 |  | 2/4 | 3 |
| 2 | 22/3 | 6/4 |  |  | 2/4 | 5 |
| 3 | 9/5 | 6/4 |  |  |  | 2 |
| 4 | 8/2 |  |  |  |  | 1 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | Р |
| 0 | /6 | /6 | /6 | /6 | 0/5 | 5 |
| 1 | 7/5 | 11/5 | 6/5 | 2/5 |  | 4 |
| 2 | 7/5 | 8/4 | 3/4 |  |  | 3 |
| 3 | 7/5 | 8/4 |  |  |  | 1 |
| 4 |  | 8/4 |  |  |  | 2 |

Алгоритм Форда-Беллмана

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 |
|  | 0 |  |  |  |  |
|  | 0 | 2 |  | 19 | 25 |
|  | 0 | 2 | 8 | 16 | 21 |
|  | 0 | 2 | 8 | 9 | 18 |
|  | 0 | 2 | 8 | 9 | 11 |

Алгоритм Флойда

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **1** | **2** | **3** | **4** | **5** |
| **1** | 0 | 2 |  | 19 | 25 |
| **2** | 2 | 0 | 6 | 14 | 27 |
| **3** | 21 | 6 | 0 | 1 | 13 |
| **4** |  | 6 | 1 | 0 | 2 |
| **5** | 7 | 9 | 6 | 2 | 0 |
|  |  |  |  |  |  |
|  | **1** | **2** | **3** | **4** | **5** |
| **1** | 0 | 2 | 8 | 16 | 25 |
| **2** | 2 | 0 | 6 | 14 | 27 |
| **3** | 8 | 6 | 0 | 1 | 13 |
| **4** | 8 | 6 | 1 | 0 | 2 |
| **5** | 7 | 9 | 6 | 2 | 0 |
|  |  |  |  |  |  |
|  | **1** | **2** | **3** | **4** | **5** |
| **1** | 0 | 2 | 8 | 9 | 21 |
| **2** | 2 | 0 | 6 | 7 | 19 |
| **3** | 8 | 6 | 0 | 1 | 13 |
| **4** | 8 | 6 | 1 | 0 | 2 |
| **5** | 7 | 9 | 6 | 2 | 0 |
|  |  |  |  |  |  |
|  | **1** | **2** | **3** | **4** | **5** |
| **1** | 0 | 2 | 8 | 9 | 11 |
| **2** | 2 | 0 | 6 | 7 | 9 |
| **3** | 8 | 6 | 0 | 1 | 3 |
| **4** | 8 | 6 | 1 | 0 | 2 |
| **5** | 7 | 8 | 3 | 2 | 0 |
|  |  |  |  |  |  |
|  | **1** | **2** | **3** | **4** | **5** |
| **1** | 0 | 2 | 8 | 9 | 11 |
| **2** | 2 | 0 | 6 | 7 | 9 |
| **3** | 8 | 6 | 0 | 1 | 3 |
| **4** | 8 | 6 | 1 | 0 | 2 |
| **5** | 7 | 8 | 3 | 2 | 0 |