<u>Лекция 8.1.</u> Мрежови модели на задачи за намиране на най-къс път и минимален скелет. Методи и алгоритми за решаването им

Съществуват линейни задачи, които могат да се решат с методите на линейното оптимиране, но са много трудоемки. Например, такъв тип задачи са:

- задачите за проектиране на автомобилни магистрали, нефто- и газопроводи и др. и то по такъв начин, че разходите за строителството на обекта да са минимални;
- задачите за проектиране на електрически съоръжения и инсталации с минимална дължина на кабелните връзки;
- задачите за определяне на най-къс път между две селища при зададена пътна мрежа

и т.н.

Трудността на прилагането на линейните методи върху този клас задачи се състои в многобройните ограничения, в които целевата функция трябва да достигне своя екстремум. За този специален клас задачи могат да се приложат *мрежови методи*, които да улеснят процеса на решение. Това са методи от *динамичното оптимиране* с итерационен подход (т.е. решаване на линейна задача с динамични методи).

8.1.1. Мрежови модели

Мрежовите модели се представят най-често чрез матрици или *графи* (*дървета*).

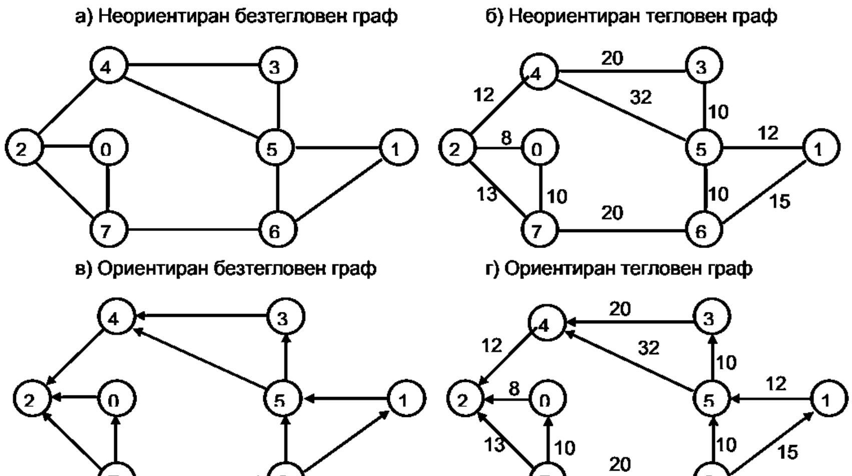
Всеки граф е изграден от *върхове* (*възли*) и *дъги* (*ребра*), които ги съединяват и могат да бъдат *насочени* (*ориентиран граф*) или *ненасочени* (*неориентиран граф*).

Два възела, съединени с ребро се наричат *съседни*. Броят на ребрата, инцидентни на даден връх се нарича *ранг на този връх*. Възел с ранг 1 е краен възел и през него не могат да преминават транзитни пътища.

От възел X до възел Y съществува връзка, ако е възможно да се предаде информация директно или чрез посреднически възли. За да се осъществи връзката е необходим *път* - крайна последователност от ребра, започвайки от възел X и завършвайки във възел Y, при което не се преминава два пъти през един и същи възел. Избран път между два възела се нарича *маршрут*. Броят на ребрата, образуващи даден път се нарича *ранг на пътя*. Минималният ранг на път е 1, а максималния – броя на възлите минус единица.

Свързан граф е този, в който кои да е два възела са свързани с поне един път. *Графът е h-свързан*, ако всеки два възела от него са свързани с най-малко h независими пътя.

На всяко ребро в графа може да се съпостави число, т.нар. *тегло*. Така графът става *тегловен*.



Фиг. 8.1.1. Видове графи

Оптимизационни задачи, които се решават с използване на мрежови модели, са:

- *Минимизация* (*минимален скелет*) *на граф;*
- *Определяне на най-къс (най-дълъг) път в граф;*
- *Определяне на максимален поток в граф;*
- *Определяне на критичен път в граф;*

8.1.2. Динамични методи

Динамичните методи са общи методи за решаване на оптимизационни задачи чрез декомпозиция на първоначалната задача на по-прости подзадачи. Определят се етапи на решението на задачата и за всеки етап се формулира по-проста оптимизационна задача. На всеки етап се определя скаларна или векторна променлива, която е оптимално решение за етапа. Резултатите от последователните етапи са свързани с рекурентен алгоритъм. Решението на първоначалната задача се получава чрез последователно решаване на задачите от отделните етапи отзад напред, като решение на задачата от последния етап. Поради това динамичните методи се определят като последователни, итерационни методи за решаване на оптимизационни задачи.

Динамичните методи се основават на *принципа на оптималността*, формулиран от Р. Белман през 1957 г.: „Една стратегия е оптимална, ако за даден етап, независимо от това какви са били решенията на предишните етапи, решенията, които ще се вземат, съставят оптимална стратегия, като се разглеждат резултатите от предишните етапи“. Накратко: „Оптималната стратегия съдържа само оптимални подстратегии“.

Елементите на модела на динамичното оптимиране са: *целева функция*, *етап*, *вариант на решение*, *състояние*, *рекурентно съотношение*.

Целевата функция е обобщен критерий за качеството на взетите решения и на постигнатите резултати при решаването на дадена задача. Тя задава начина, по който сложната задача се декомпозира на по-прости.

- При *адитивната декомпозиция* общият резултат е сума от резултатите на отделните етапи. Примери на адитивни целеви функции са общият доход или общите загуби, реализирани на всеки етап.
- При *мултипликативната декомпозиция* общият резултат е произведение от резултатите на отделните етапи. Пример на мултипликативна функция е надеждността на система от последователно съединени елементи.

Етапът съответства на подзадача. Неговият принос в общия резултат се описва с целева функция, която е адитивна или мултипликативна част на общата целева функция. Той може да представлява интервал от време или да не е свързан с времето.

Вариантите на решение определят една или друга стойност на целевата функция. Те са съвкупност от стойностите на променливите. Преминавайки от етап на етап, стойностите на част от променливите се променят, давайки поредния вариант.

Изборът на *променливите на състоянието* x_j при решаването на задачи на динамичното оптимиране е най-важен. Те осигуряват връзка между етапите, при които вариантите на решението се получават в резултат на оптимизация на задачата. При определянето на оптималните решения на всеки етап се използват само допустими варианти и не се проверява влиянието на текущия избор върху решенията, избрани на предишните етапи.

В динамичното оптимиране се ползват *обратно* или *право рекурентни съотношения*.

При *детерминирани задачи* на динамичното оптимиране състоянието на следващия етап се определя напълно от състоянието на сегашния етап. Такава задача е и *задачата за раницата* или *задача за товара с максимална полезност*.

8.1.3. Алгоритми за решаване на задачи за намиране на минимален скелет на неориентиран граф

Задача за намиране на минимален скелет

За дадена мрежа $M = \{V, D, C\}$ (фиг. 2),
където:

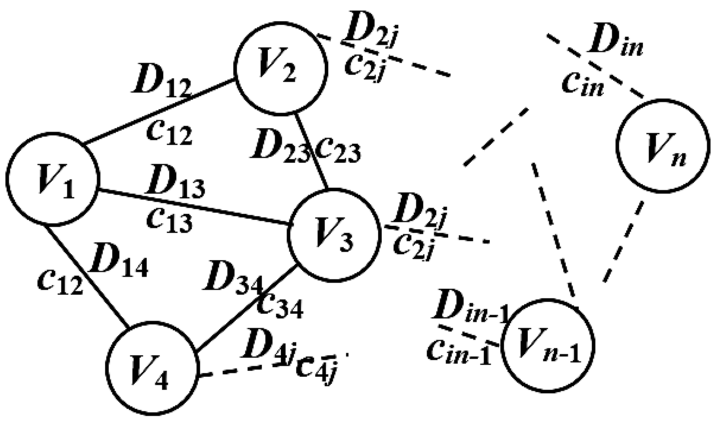
V е множеството от върховете (възлите);

D е множеството от дъгите;

C е множеството от теглата на дъгите на мрежата,

да се намери скелет $\hat{G}_j = \{V, \hat{D}_j\}$ на графа $G = \{V, D\}$, за който сумата от характеристиките на скелетните дъги (теглата на дъгите) да е минимална, т.е.

$$\min F_j = \sum_{k=1}^{n-1} \{c_k |\ D_k \epsilon \hat{D}_j\}$$



Фиг. 8.1.2. Мрежа $M = \{V, D, C\}$

Примери на задачи за намиране на минимален скелет на мрежа:

- задачи за проектиране на автомобилни магистрали, нефто- и газопроводи и др. и то по такъв начин, че разходите за строителството на обекта да са минимални;
- задачи за проектиране на електрически съоръжения и инсталации с минимална дължина на кабелните връзки,

където възлите са статични обекти, а дъгите могат да бъдат дължина на пътя между обектите, разходи, цена и т.н.

Алгоритъм на Прим

Стъпка 1. Множеството на възлите V на графа $G = \{V, D\}$ се разделят на две, както следва $V_1' = \{V_i\}$ и $V_1'' = V \backslash V_1'$.

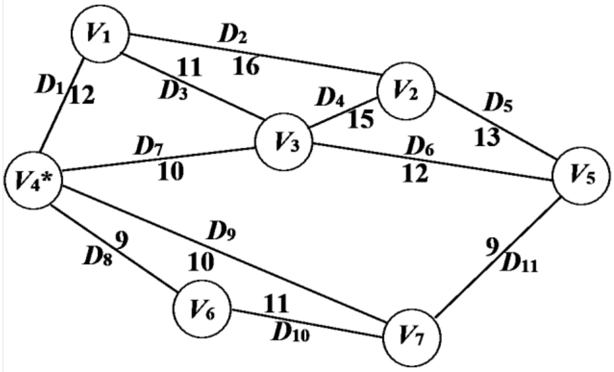
Стъпка 2. Определя се коцикъла на V_1' и между дъгите от коцикъла се избира тази с минимално тегло, която е $c_k \leftrightarrow D_k = \{V_i, V_j\}$.

Стъпка 3. Отново V се разделя на две по следния начин: $V_2' = \{V_i, V_j\}$ и $V_2'' = V \backslash V_2'$, т.е. към множеството V_2' се включва и другия край на избраната дъга с минимално тегло. Между дъгите от коцикъла на V_2' се избира тази с минимално тегло и т.н. до коцикъла на V_{n-1}'.

Забележка. Ако изборът на минимално тегло не е еднозначен, то се избира една произволна дъга с минимално тегло.

Пример

Да се намери минимален скелет и стойността му за зададения граф с начален възел V_4 с алгоритъма на Прим:



Решение

1. Определяме множествата V_1', състоящо се само от началния възел V_1'', който съдържа всички останали възли:

$$V_1' = \{V_4\}, V_1^{''} = \{V_1; V_2; V_3; V_5; V_6; V_7\}$$

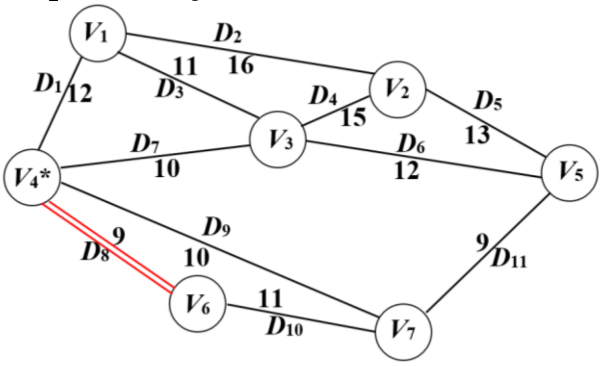
Коцикълът на V_1' при това деление се състои от всички дъги, чийто краища лежат в двете множества V_1' и V_1'':

$$CC^1 = \{D_1; D_7; D_9; D_8\}$$

Теглата на тези дъги са:

$$c_1 = 12, c_7 = 10, c_9 = 10, c_8 = 9$$

Измежду тези дъги избираме тази с минимално тегло. Това е дъга D_8 с тегло $c_8 = 9$. Възелът, който е инцидентен с тази дъга и лежи в множеството V_1'' е V_6, така че той се включва в множеството V_1', а дъга D_8 става част от минималния скелет, който търсим.



2. Новото деление е:

$$V_2' = \{V_4; V_6\}, V_2'' = \{V_1; V_2; V_3; V_5; V_7\}$$

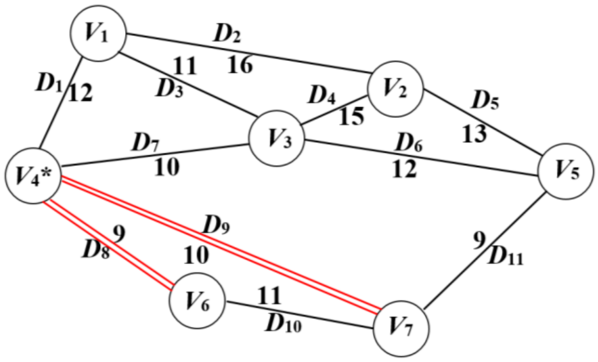
Коцикълът на V_2' е:

$$CC^2 = \{D_1; D_7; D_9; D_{10}\}$$

В този коцикъл има две дъги с равни минимални тегла - D_7 и D_9, като:

$$c_7 = c_9 = 10$$

Избираме произволна от тях, например D_9, която става част от минималния скелет, а другия край на тази дъга – възел V_7 присъединяваме към множеството V_2'.



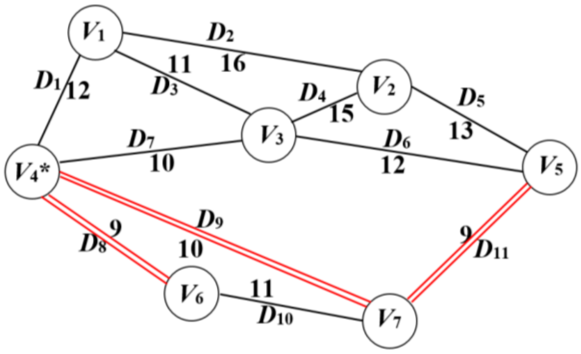
3.

$$V_3' = \{V_4; V_6; V_7\}, V_3'' = \{V_1; V_2; V_3; V_5\}$$
$$CC^3 = \{D_1; D_7; D_{11}\}$$

Дъгата с минимално тегло, участваща в коцикъла CC^3 е D_{11} с тегло:

$$c_{11} = 9$$

Фиксираме я като част от минималния скелет, а възел V_5 – инцидентен с нея и го включваме в множеството V_3'.



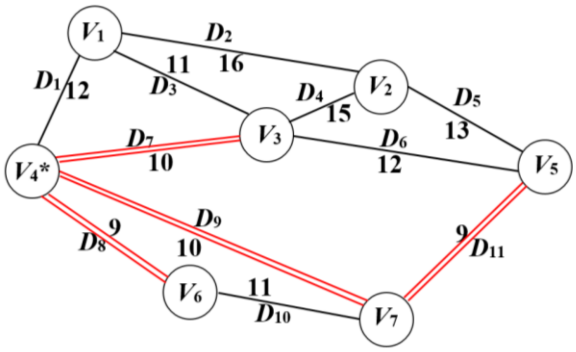
4.

$$V_4' = \{V_4; V_6; V_7; V_5\}, V_4'' = \{V_1; V_2; V_3\}$$
$$CC^4 = \{D_1; D_7; D_6; D_5\}$$

Дъгата D_7 има минимално тегло:

$$c_7 = 10$$

Фиксираме я в минималния скелет, а другия край – възел V_3 го присъединяваме към множеството V_4'.



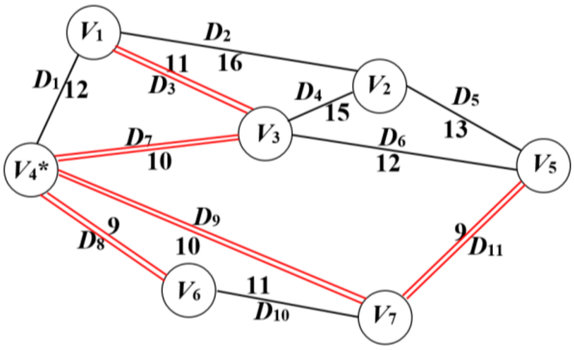
5.

$$V_5' = \{V_4; V_6; V_7; V_5; V_3\}, V_5'' = \{V_1; V_2\}$$
$$CC^5 = \{D_1; D_3; D_4; D_5\}$$

Измежду дъгите, участващи в този коцикал, тази с минимално тегло е D_3 с

$$c_3 = 11$$

Включваме дъгата D_3 в минималния скелет, а другия ѝ край – възел V_1 става част от множеството V_5'.



6.

$$V_6' = \{V_4; V_6; V_7; V_5; V_3; V_1\}, V_6'' = \{V_2\}$$
$$CC^6 = \{D_2; D_4; D_5\}$$

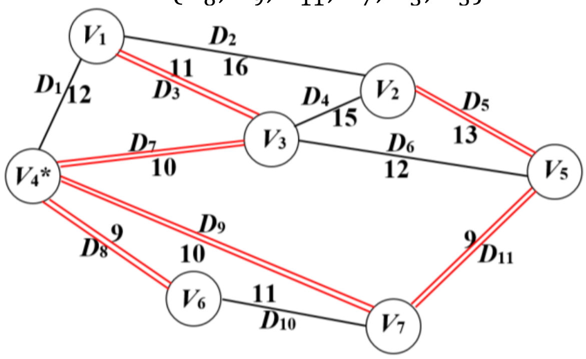
Дъга D_5 има минимално тегло

$$c_5 = 13$$

Тази дъга става част от минималния скелет, а другия ѝ край е възел V_2, който се включва в множеството V_6', а множеството V_6'' остава празно.

7. Следователно, минималният скелет $\hat{G} = \{V, \hat{D}\}$ е намерен и той включва дъгите

$$\hat{D} = \{D_8; D_9; D_{11}; D_7; D_3; D_5\}$$



Стойността му се намира кето се сумират теглата на дъгите, участващи в него:

$$F_{min} = c_8 + c_9 + c_{11} + c_7 + c_3 + c_5 = 9 + 10 + 9 + 10 + 11 + 13 = 62$$

Задача

Да се определят ребрата, които свързват всички възли на графа и имат минимално сумарно тегло.

Алгоритъм на Белман

Решението на задачата определя минимален граф, в който между всеки два възела съществува верига и в нея няма цикли.

Итерационната процедура е следната:

Стъпка 1. В началото всички възли на графа са несвързани (неприсъединени).

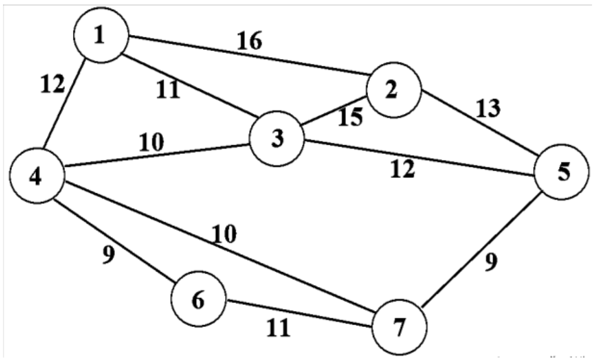
Стъпка 2. Сред тях се избира начален възел, които се свързва с най-близкия възел.

Стъпка 3. Сред неприсъединените възли се определя възелът, които е разположен най-близо до някои от вече присъединените възли и тези два възела се свързват. Процесът се повтаря до изчерпване на неприсъединените възли.

Забележка. Ако на някоя итерация има няколко неприсъединени възли, които са на равно най-малко разстояние до някои от вече присъединените възли, оптималното решение може да не е единствено.

Пример

Да се намери минимален скелет и стойността му за зададения граф с алгоритъма на Белман:



Решение

Стъпка 1. В началото всички възли на графа са несвързани (неприсъединени).

Построяваме таблица, в което ще отчитаме постепенното присъединяване на възлите:

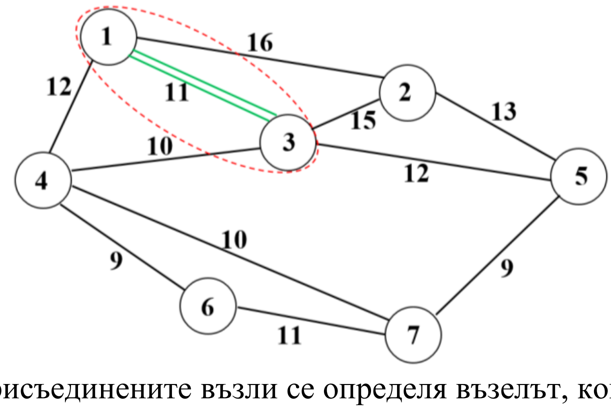
Неприсъединени възли	**Присъединени възли**
1, 2, 3, 4, 5, 6, 7	
Дължина на дъгите от минималния скелет	

Стъпка 2. Сред възлите се избира начален възел, които се свързва с най-близкия възел.

1. Например, нека този възел е 1 и измежду този възел и свързаните с него неприсъединени възли 4, 3 и 2, най-близкия до него възел е възел 3. Тогава в таблицата маркираме:

Неприсъединени възли	**Присъединени възли**
~~1~~, 2, ~~3~~, 4, 5, 6, 7	**1, 3**
Дължина на дъгите от минималния скелет	11

На графа правим следните означения:



Стъпка 3. Сред присъединените възли се определя възелът, които е разположен най-близо до някои от неприсъединените възли по следния начин:

2. Съществуват от присъединения възел 1 към неприсъединените възли 4 и 2 дъги съответно с дължина 12 и 16, а от присъединения възел 3 към неприсъединените възли 4, 2 и 5 дъги съответно с дължини 10, 15 и 12. Тогава:

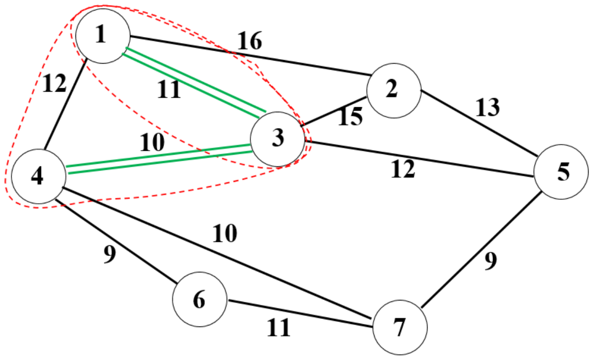
$$min\{12,16,10,15,12\} = 10$$

Минималната дължина между присъединен и неприсъединен възел е 12 и тя е между възли 1 и 4 и между възли 3 и 5 и не е еднозначно определена, т.е. можем да изберем една от двете дъги. Нека например това е дъгата между 1 и 4.

Тогава в таблицата маркираме:

Неприсъединени възли	**Присъединени възли**
~~1~~, 2, ~~3~~, ~~4~~, 5, 6, 7	1, 3, 4
Дължина на дъгите от минималния скелет	11+10

На графа правим следните означения:

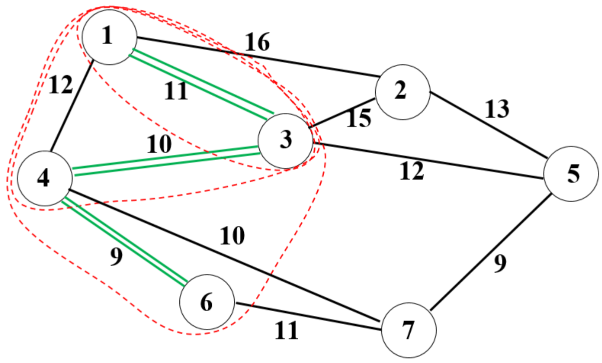


3. От присъединените възли 1, 3 и 4 съществуват дъги към неприсъединените възли 2, 5, 6 и 7. Тогава минималната дължина между присъединен и неприсъединен възел е:

$$min\{16,15,12,10,9\} = 9$$

и това е между присъединен възел 4 и неприсъединен възел 6. Присъединяваме възел 6:

Неприсъединени възли	**Присъединени възли**
~~1~~, 2, ~~3~~, ~~4~~, 5, ~~6~~, 7	1, 3, 4, 6
Дължина на дъгите от минималния скелет	11+10+9

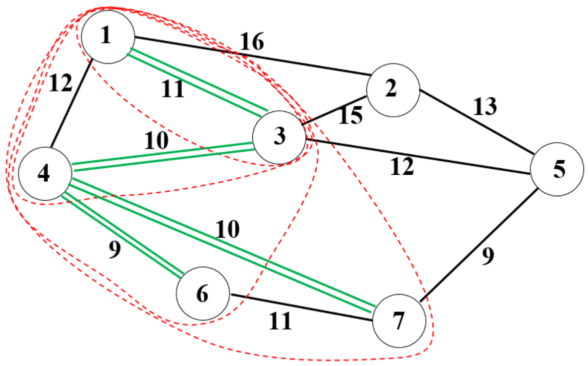


4. От присъединените възли 1, 3, 4 и 6 съществуват дъги към неприсъединените възли 2, 5 и 7. Тогава минималната дължина между присъединен и неприсъединен възел е:

$$min\{11,10,12,15,16\} = 10$$

и това е между присъединен възел 4 и неприсъединен възел 7. Присъединяваме възел 7:

Неприсъединени възли	**Присъединени възли**
~~1~~, 2, ~~3~~, ~~4~~, 5, ~~6~~, ~~7~~	1, 3, 4, 6, 7
Дължина на дъгите от минималния скелет	11+10+9+10

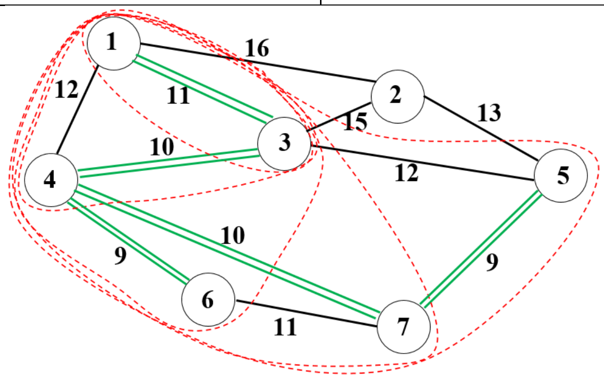


5. От присъединените възли 1, 3, 4, 6 и 7 съществуват дъги към неприсъединените възли 2 и 5. Тогава минималната дължина между присъединен и неприсъединен възел е:

$$min\{9,12,15,16\} = 9$$

и това е между присъединен възел 7 и неприсъединен възел 5. Присъединяваме възел 5:

Неприсъединени възли	**Присъединени възли**
~~1~~, 2, ~~3~~, ~~4~~, ~~5~~, ~~6~~, ~~7~~	**1, 3, 4, 6, 7, 5**
Дължина на дъгите от минималния скелет	11+10+9+10+9

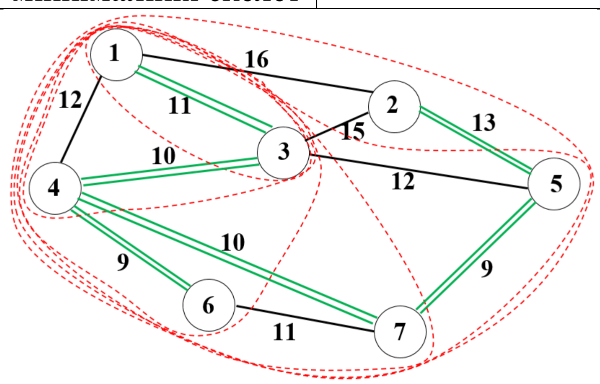


6. От присъединените възли 1, 3, 4, 6, 7 и 5 съществуват дъги към неприсъединения възел 2. Тогава минималната дължина между присъединен и неприсъединен възел е:

$$min\{16,15,13\} = 13$$

и това е между присъединен възел 5 и неприсъединен възел 2. Присъединяваме възел 2:

Неприсъединени възли	**Присъединени възли**
~~1~~, ~~2~~, ~~3~~, ~~4~~, ~~5~~, ~~6~~, ~~7~~	**1, 3, 4, 6, 7, 5, 2**
Дължина на дъгите от минималния скелет	11+10+9+10+9+13



7. Изчерпали сме неприсъединените възли и сме намерили минималния скелет на графа и $F_{min} = 11 + 10 + 9 + 10 + 9 + 13 = 62$