ЗВЯЗНІСТЬ ГРАФА.
МІНІМАЛЬНІ ШЛЯХИ В ЗВАЖЕНИХ ОРГРАФАХ

1. Зв'язність графа

Означення. ***Граф*** G *називається* ***зв'язним****, якщо будь-яка пара його вершин може бути з'єднана маршрутом.*

Означення. *Орграф* $\vec{G}$ *називається* ***зв'язним****, якщо існують шляхи для всіх пар різних вершин графа.*

Для орграфів поняття зв'язності є більше змістовним, чим для неорієнтованих графів. Розрізняють три важливих типи зв'язності орграфа:

а) орграф $\vec{G}$ *називається* ***сильно зв'язни****м*, якщо для кожної пари різних вершин v_i, v_j з V існує шлях (орієнтований ланцюг) з v_i в v_j і з v_j в v_i.

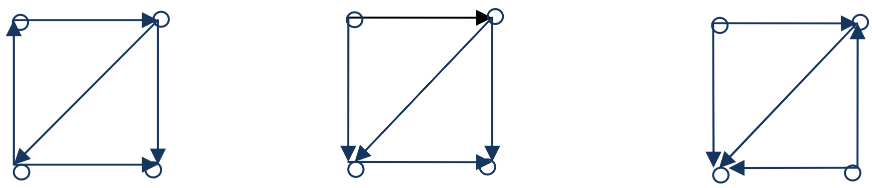
б) орграф $\vec{G}$ *називається* ***односторонньо зв'язни****м*, якщо для кожної пари різних вершин v_i, v_j з V існує шлях з v_i ***в*** v_j ***або з*** v_j ***в*** v_i**.**

в) орграф $\vec{G}$ *називається* ***слабко зв'язни****м*, якщо граф, отриманий з $\vec{G}$ скасуванням орієнтації є зв'язним.

Очевидно, що справедливі наслідки:

$\vec{G}$ сильно зв'язний $\Rightarrow \vec{G}$ односторонньо зв'язний $\Rightarrow$ $\vec{G}$ слабко зв'язний.

Приклад 1.



Сильна, одностороння і слабка зв'язність.

Означення. ***Матрицею досяжності*** *графа* $G=(V,E)$ *називається матриця* $R(G)=\left\{r_{ij}, i,j=\overline{1,n}\right\}$, *де*

$$r_{ij}=\begin{cases}1, \text{ якщо існує маршрут з вершини } v_i \text{ у вершину } v_j;\\ 0, \text{ в противному випадку.}\end{cases}$$

Граф $G=(V,E)$ є зв'язним тоді й тільки тоді, коли для всіх $i,j=\overline{1,n}$ $r_{ij}=1$. (матриця досяжності заповнена одиницями).

<u>Означення.</u> *Нехай* X *– деяка матриця. Булевим відображенням для матриці* X *називається відображення* $X \to B(X) = \begin{cases} B(x_{ij}) = 1, \text{ якщо } x_{ij} \neq 1; \\ B(x_{ij}) = 0, \text{ якщо } x_{ij} = 0. \end{cases}$

<u>Твердження.</u> ***Матриця досяжності*** *графа* $G = (V, E)$ *є булеве відображення матриці* $E + A + A^2 + \ldots + A^n$*, тобто*

$$R(G) = B\left(E + A + A^2 + \ldots + A^n\right).$$

де n *– число вершин,* E *– одинична матриця.*

Матриця досяжності $R(\vec{G})$ орграфа $\vec{G}$ визначається аналогічно.

У термінах матриці зв'язності $R(\vec{G})$ орграф $\vec{G}$ сильно зв'язний тоді і тільки тоді, коли $r_{ij} = 1$ для всіх $i, j = \overline{1,n}$; $\vec{G}$ односторонньо зв'язний тоді й тільки тоді, коли $r_{ij} = 1$ або $r_{ji} = 1$ для всіх $i, j = \overline{1,n}$.

Властивість зв'язности можна розглянути як бінарне відношення на множині вершин V, яке:

а) рефлексивне: вершина v_i зв'язана сама із собою;

б) симетричне: якщо вершина v_i зв'язана з вершиною v_j, то й вершина v_j зв'язана з вершиною v_i

в) транзитивне: якщо вершина v_i зв'язана з вершиною v_j, і v_j зв'язана з вершиною v_k, то вершина v_i зв'язана з вершиною v_k.

Відношення зв'язності є відношенням еквівалентності, тобто воно розбиває множину V вершин графа на класи V_i, які попарно не перерізаються. Оскільки кожна множина V_i – множина зв'язаних вершин, а вершини з різних множин V_i не зв'язані, то маємо розбиття графа G на частини, які не перерізаються і кожна частина – зв'язна.

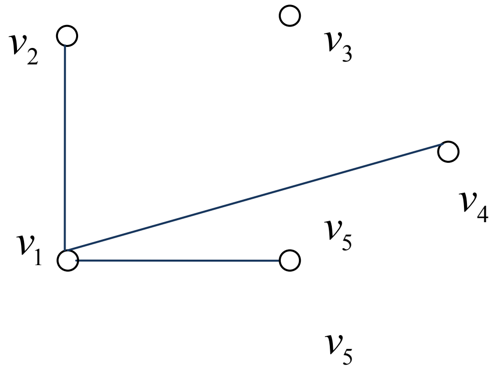
<u>Означення.</u> *Нехай* $\{V_i,\ i \in N_p\}$ *– розбиття графа* $G = (V, E)$*, обумовлене відношенням зв'язності. Число* p *називається* ***числом зв'язності*** *графа* G.

<u>Означення</u>. ***Компонентами зв'язності*** *графа* G *називаються підграфи* (V_i, E_i) *графа, породжені класами еквівалентності.*

<u>Означення</u>. ***Матрицею зв'язності*** *графа* $G = (V, E)$ *називається матриця* $S(G) = \{s_{ij}, i, j = \overline{1,n}\}$*, де*

$$s_{ij} = \begin{cases} 1, \text{ якщо вершини } v_i \text{ і } v_j \text{ належать одній компоненті зв'язності;} \\ 0, \text{ в противному випадку.} \end{cases}$$

Приклад 2. В графа G дві компоненти зв'язності:



$K_1 = \{v_1, v_2, v_4, v_5\}$;

$K_2 = \{v_3\}$.

Матриця зв'язності цього графа має вигляд:

$$S(G) = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \end{pmatrix}$$

1-й, 2-й, 4-й, 5-й рядки матриці $S(G)$ однакові.

Компоненти зв'язності графа G визначаються за допомогою його матриці досяжності $R(G)$. Для виявлення компонентів зв'язності графа G треба побудувати матрицю $R(G) \otimes R^T(G)$, де $\otimes$ – поелементне множення матриць. Якщо отримана матриця має блоково-діагональний вигляд, то кожен блок визначає одну компоненту зв'язності.

Число елементів в компоненті зв'язності, в яку входить вершина v_i, легко визначити, підносячи матрицю досяжності $R(G)$ до квадрату. Діагональні елементи $(r^2)_{ii}$ матриці $R^2(G)$ показують число елементів в тій компоненті зв'язності, в яку входить вершина v_i.

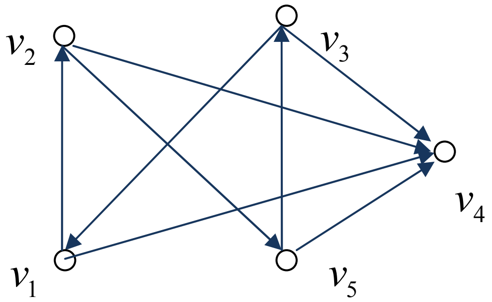
Відзначимо, що *для орграфів відношення зв'язності не є відношенням еквівалентності* на множині вершин V і, отже, не здійснює розбиття множини V.

Означення. ***Компонентами сильної зв'язності*** *орграфа* $\vec{G}$ *називаються його максимальні сильно зв'язні підграфи.*

Кожна вершина орграфа належить тільки одній компоненті сильної зв'язності. Якщо вершина не зв'язана з іншими, то вважають, що вона сама утворює компоненту сильної зв'язності.

Матриця сильної зв'язності орграфа $\vec{G}$ визначається аналогічно матриці зв'язності неорієнтованого графа. Компоненти сильної зв'язності визначаються за допомогою його матриці досяжності $R(\vec{G})$ так само, як і для неорієнтованого графа.

Приклад 3. В орграфа $\vec{G}$ дві компоненти сильної зв'язності:



$$K_1 = \{v_1, v_2, v_3, v_5\};$$

$$K_2 = \{v_4\}.$$

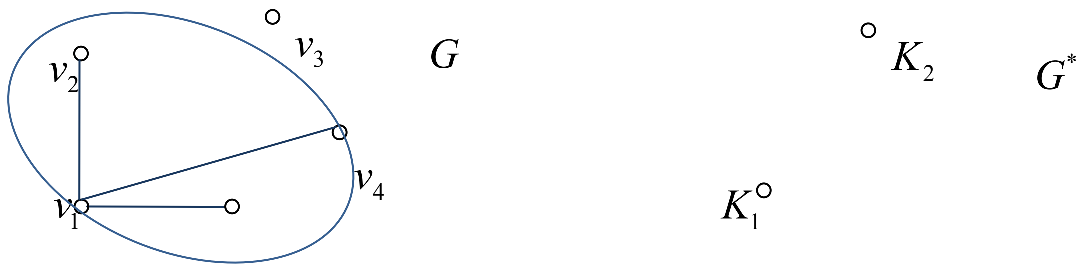
Матриця сильної зв'язності цього графа має вигляд:

$$S(\vec{G}) = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 \end{pmatrix}$$

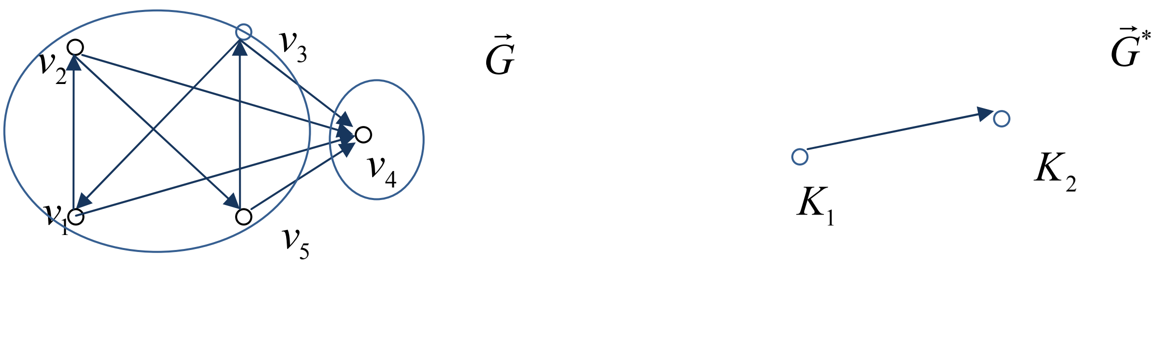
1-й, 2-й, 3-й, 5-й рядки матриці $S(\vec{G})$ однакові.

Означення. ***Конденсацією*** *G^* ($\vec{G}^*$) графа G (орграфа $\vec{G}$) (або фактор-графом) називається граф, який отриманий стягуванням в одну вершину кожної компоненти зв'язності (сильної зв'язності) графа G (орграфа $\vec{G}$).*

Приклад 4. Конденсацією графа з прикладу 2 є граф



Конденсацією орграфа з прикладу 3 є граф



2. Мінімальні шляхи в зважених орграфах

<u>Означення</u>. *Орграф* $\vec{G} = \left(V, \vec{E}\right)$ *називається* ***зваженим****, якщо кожній дузі* $\left(v_i, v_j\right)$ *зіставлене деяке число* $c\left(v_i, v_j\right)$*, яке називається її* ***довжиною*** *(або* ***вагою****, або* ***вартістю****).*

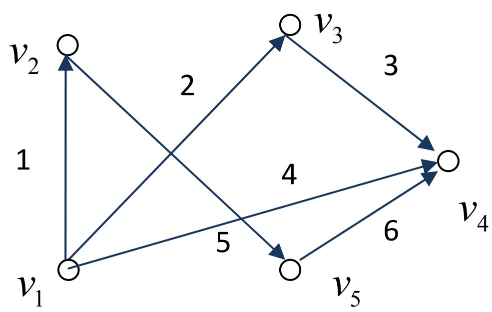
<u>Означення</u>. ***Довжиною*** *(або* ***вагою****, або* ***вартістю****)* ***шляху*** s*, який складається з деякої послідовності дуг* $\left(v_i, v_j\right)$*, називається число* $l\left(s\right)$*, яке дорівнює сумі довжин дуг, що входять в цей шлях, тобто*

$$l\left(s\right) = \sum_{\left(v_i, v_j\right) \in s} c\left(v_i, v_j\right).$$

<u>Означення.</u> ***Матрицею довжин дуг*** *або* ***матрицею вагів*** *орграфа* $\vec{G} = \left(V, \vec{E}\right)$ *називається матриця* $C\left(\vec{G}\right) = \left\{c_{ij}, i, j = \overline{1, n}\right\}$*, де*

$$c_{ij} = \begin{cases} c\left(v_i, v_j\right), \text{ ÿêù î ³ñíóº äóãà ç âóçëà } v_i \text{ ó âóçî ë } v_j; \\ \infty, \text{ â ïðî òèâíî ì ó âèïàäêó.} \end{cases}$$

***<u>Приклад</u>* 5.** Для зваженого орграфа, зображеного на малюнку матриця $C\left(\vec{G}\right)$ має вигляд:



$$C\left(\vec{G}\right) = \begin{pmatrix} \propto & 1 & 2 & 4 & \propto \\ \propto & \propto & \propto & \propto & 5 \\ \propto & \propto & \propto & 3 & \propto \\ \propto & \propto & \propto & \propto & \propto \\ \propto & \propto & \propto & 6 & \propto \end{pmatrix}$$

Довжина шляху $s = \left(v_1, v_2, v_5, v_4\right)$ дорівнює $l\left(s\right) = 1 + 5 + 6 = 12$.

<u>Означення.</u> ***Відстанню*** $d\left(v_i, v_j\right)$ *між вузлами* v_i *і* v_j *називається довжина найкоротшого шляху між цими вузлами.*

<u>Зауваження</u>. Відмінність від неорієнтованих графів полягає в тому, що, взагалі кажучи, $d\left(v_i, v_j\right) \neq d\left(v_j, v_i\right)$.

При розв’язуванні багатьох практичних задач виникає необхідність пошуку мінімального шляху між двома довільними вузлами (шляху з найменшою сумою довжин дуг). Для знаходження мінімального шляху між двома довільними вузлами існують певні алгоритми. Наприклад, алгоритм Дейкстри, алгоритм Форда-Беллмана.

Розглянемо пошук мінімального шляху між двома різними вузлами на прикладі алгоритму Дейкстри (Е́дсгер Ві́бе Де́йкстра (Edsger Wybe Dijkstra) (1930–2002) – нідерландський вчений, математик-програміст). Цей алгоритм дозволяє знайти відстань від заданого вузла v_s до всіх інших вузлів зваженого орграфа у випадку, коли всі $c_{ij} \geq 0$.

Алгоритм Дейкстри

Крок 0. Всі вузли помічаються: стартовий вузол v_s отримує постійну мітку 0, всі інші вузли отримують тимчасові мітки ∞.

Крок i для будь-якого $i > 0$. Для всіх вузлів v_j з тимчасовими мітками, в які заходять дуги з вузла v_i з постійною міткою, оновлюємо мітки. За нову мітку обираємо мінімум із старої мітки і суми мітки вузла v_i і відстані між v_i і v_j. Серед вузлів v_j з тимчасовими мітками, знаходимо вузол з найменшою міткою. Мітку цього вузла проголошуємо постійною. (При чисельному виконанні алгоритму вручну мітку підкреслюємо).

Якщо всі вузли отримали постійні мітки, закінчити роботу алгоритму. Отримані постійні мітки дають відстані від стартового вузла v_s до всіх інших вузлів орграфа.

Якщо залишилися вузли з тимчасовими мітками, повторити крок ***i***.

Таким чином, спочатку виконується ініціалізація (крок 0), а потім виконуються кроки 1,2,3,…, доти, поки є тимчасові мітки.

Введемо позначення:

$l\left(v_j\right)$ – тимчасова мітка вузла v_j;

$l^*\left(v_j\right)$ – постійна мітка вузла v_j;

$G\left(v_i\right)$ – множина вузлів, в які заходять дуги з вузла v_i з постійною міткою.

Перепишемо алгоритм Дейкстри в більш зручному для застосувань вигляді:

Крок 0. $l^*\left(v_s\right) = 0$, $\forall v_j \neq v_s \;\; l\left(v_j\right) = \infty$.

Крок i. $G\left(v_i\right) = \left\{v_j : \exists\left(v_i, v_j\right)\right\}$;

$$\forall v_j \in G\left(v_i\right) \quad l\left(v_j\right) = \min\left\{l\left(v_j\right), l\left(v_i\right) + d\left(v_i, v_j\right)\right\},$$

$$l^*\left(v_j\right) = \min_{v_j}\left\{l\left(v_j\right)\right\}.$$

$$\forall v_j \in V \; l^*\left(v_j\right) = d\left(v_s, v_j\right).$$

Розглянемо роботу алгоритму Дейкстри на прикладі.

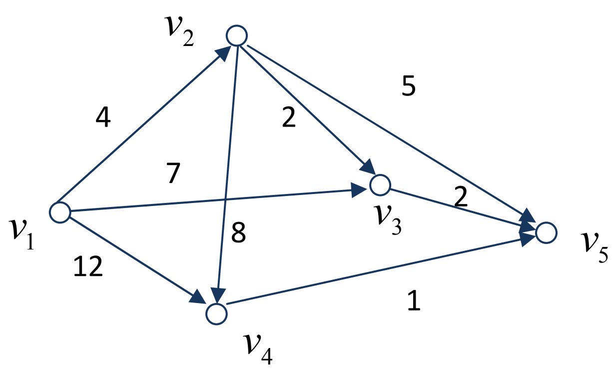
<u>Приклад 1</u>. Користуючись алгоритмом Дейкстри, знайти в зваженому орграфі $\vec{G}$, заданому матрицею довжин дуг:

$$C\left(\vec{G}\right) = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} \!\!\! \begin{array}{c} \begin{array}{ccccc} v_1 & v_2 & v_3 & v_4 & v_5 \end{array} \\ \begin{pmatrix} \propto & 4 & 7 & 12 & \propto \\ \propto & \propto & 2 & 8 & 5 \\ \propto & \propto & \propto & \propto & 2 \\ \propto & \propto & \propto & \propto & 1 \\ \propto & \propto & \propto & \propto & \propto \end{pmatrix} \end{array}$$

1) відстані від вузла v_1 до всіх інших вузлів зваженого орграфа $\vec{G}$:

2) мінімальний шлях від вузла v_1 до вузла v_5.

<u>Розв'язання</u>. 1) Намалюємо для наочності геометричне зображення графа



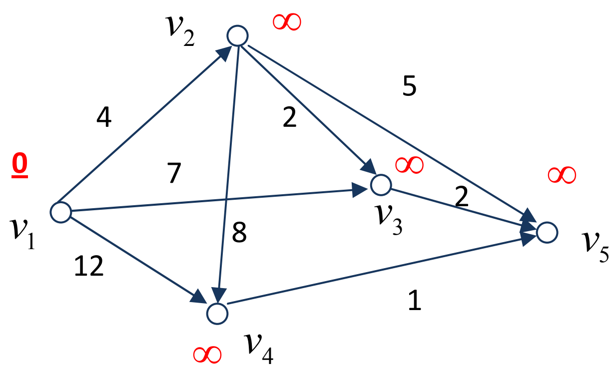
Протокол роботи алгоритму будемо оформлювати у вигляді таблиці, яку побудуємо в процесі розв'язування.

Крок 0. Всі вузли помічаються: стартовий вузол v_1 отримує постійну мітку 0, всі інші вузли отримують тимчасові мітки ∞:

$$l^*\left(v_1\right) = 0, \; \forall v_j \neq v_s \;\; l\left(v_j\right) = \infty.$$

Ці значення занесемо в нульовий рядок таблиці.

Намалюємо для наочності геометричне зображення графа на кроці 0:



Крок 1. Для вузлів v_2, v_3, v_4 з тимчасовими мітками, в які заходять дуги з вузла v_1 з постійною міткою, оновлюємо мітки:

$G(v_1) = \{v_2, v_3, v_4\}$;

$l(v_2) = \min\{\infty, 0+4\} = 4_1$;

$l(v_3) = \min\{\infty, 0+7\} = 7_1$;

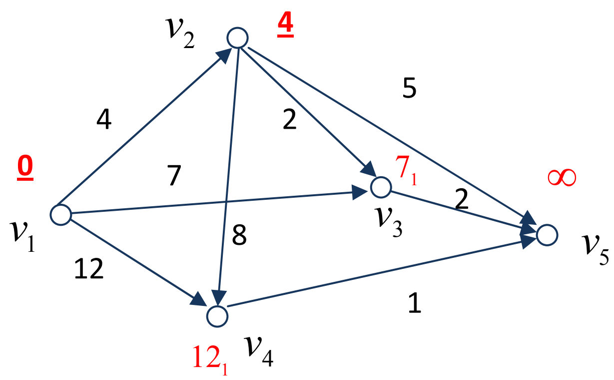
$l(v_4) = \min\{\infty, 0+12\} = 12_1$

$l^*(v_2) = \min\{4_1, 7_1, 12_1\} = 4$.

Постійну мітку отримує вузол v_2.

Ці значення занесемо в перший рядок таблиці.

Намалюємо для наочності геометричне зображення графа на кроці 1:



Крок 2. Для вузлів v_3, v_4, v_5 з тимчасовими мітками, в які заходять дуги з вузла v_2 з постійною міткою, оновлюємо мітки:

$G(v_2) = \{v_3, v_4, v_5\}$;

$l(v_3) = \min\{7_1, 4+2\} = 6_2$;

$l(v_4) = \min\{12_1, 4+8\} = 12_2$;

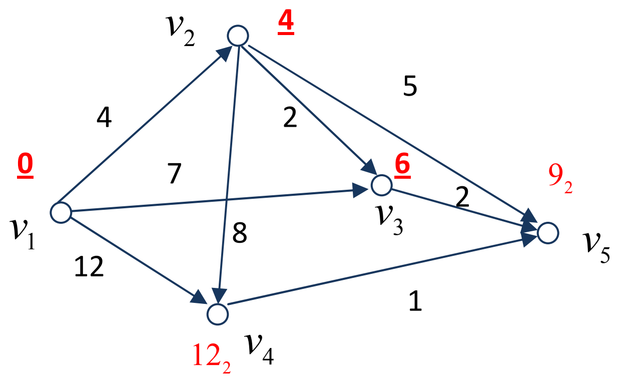
$l(v_5) = \min\{\infty, 4+5\} = 9_2$;

$l^*(v_3) = \min\{6_2, 12_2, 9_2\} = 6$.

Постійну мітку отримує вузол v_3.

Ці значення занесемо в другий рядок таблиці.

Намалюємо для наочності геометричне зображення графа на кроці 2:



Крок 3. Для вузла v_5 з тимчасовою міткою, в який заходить дуга з вузла v_3 з постійною міткою, оновлюємо мітку:

$G(v_3) = \{v_5\}$;

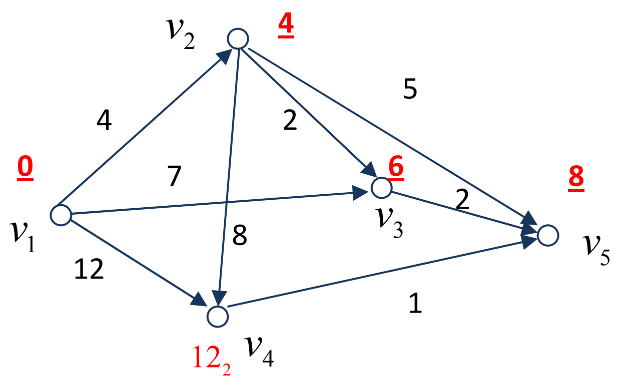
$l(v_5) = \min\{9_2, 6+2\} = 8_3$

$l^*(v_5) = \min\{8_3\} = 8$.

Постійну мітку отримує вузол v_5.

Ці значення занесемо в третій рядок таблиці.

Намалюємо для наочності геометричне зображення графа на кроці 3:



Крок 4. З вузла v_5 з постійною міткою не входить жодна дуга.

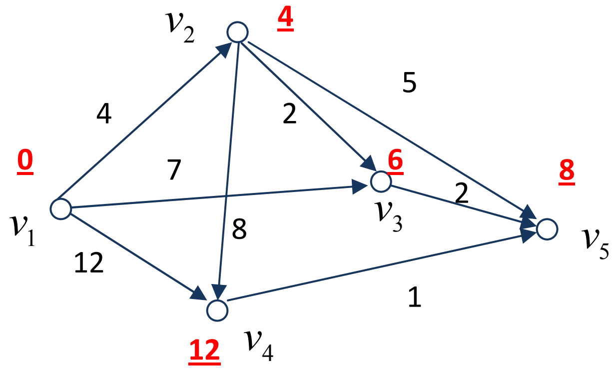
$G(v_5) = \varnothing$;

$l^*(v_4) = 12$.

Постійну мітку отримує вузол v_4.

Ці значення занесемо в четвертий рядок таблиці.

Намалюємо для наочності геометричне зображення графа на кроці 4:



Таблиця.

Вузли / Кроки	v_1	v_2	v_3	v_4	v_5
0	$\underline{0}$	∞	∞	∞	∞
1	$\underline{0}$	$\underline{4}$	7_1	12_1	∞
2	$\underline{0}$	$\underline{4}$	$\underline{6}$	12_2	9_2
3	$\underline{0}$	$\underline{4}$	$\underline{6}$	12_2	$\underline{8}$
4	$\underline{0}$	$\underline{4}$	$\underline{6}$	$\underline{12}$	$\underline{8}$

Всі вузли отримали постійні мітки. Алгоритм завершено:

$d(v_1, v_2) = 4$, $d(v_1, v_3) = 6$, $d(v_1, v_4) = 12$, $d(v_1, v_5) = 8$.

2) Щоб знайти мінімальний шлях від вузла v_1 до вузла v_5, використаємо тимчасові індекси в постійних міток: у вузла v_5 – 8_3, у вузла v_3 – 6_2, у вузла v_2 – 4_1. Отже, мінімальний шлях з v_1 до v_5 є v_1, v_2, v_3, v_5. Його довжина

$$l(s) = \sum_{(v_i, v_j) \in s} c(v_i, v_j) = c(v_1, v_2) + c(v_2, v_3) + c(v_3, v_5) = 4 + 2 + 2 = 8.$$