

Міністерство освіти і науки України  
Харківський національний університет імені В.Н. Каразіна  
Кафедра штучного інтелекту та програмного забезпечення

**“ЗАТВЕРДЖУЮ”**  
Проректор  
з науково-педагогічної роботи  
Антон ПАНТЕЛЕЙМОНОВ  
«\_\_\_\_\_» \_\_\_\_\_ 2020 р.



**РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

**Теорія алгоритмів**

рівень вищої освіти	перший (бакалаврський) рівень
галузь знань	15 Автоматизація та приладобудування
спеціальність	151 Автоматизація та комп'ютерно-інтегровані технології
освітня програма	Автоматизація та комп'ютерно-інтегровані технології
вид дисципліни	обов'язкова
факультет	комп'ютерних наук

Програму рекомендовано до затвердження Вченою радою факультету комп'ютерних наук  
«31» серпня 2020 року, протокол № 12


**РОЗРОБНИКИ ПРОГРАМИ:**

кандидат технічних наук, доцент кафедри штучного інтелекту та програмного забезпечення **Олешко Олег Іванович**,

кандидат технічних наук, доцент кафедри моделювання систем та технологій **Щебенюк Володимир Сергійович**.

Програму схвалено на засіданні кафедри штучного інтелекту та програмного забезпечення  
Протокол від «31» серпня 2020 року № 1

Завідувач кафедри штучного інтелекту та програмного забезпечення

  
Володимир КУКЛІН

Програму погоджено з гарантом освітньої програми 151 «Автоматизація та комп'ютерно-інтегровані технології»

Гарант освітньої програми 151 «Автоматизація та комп'ютерно-інтегровані технології»

  
Дмитро ЛАБЕНКО

Програму погоджено методичною комісією факультету комп'ютерних наук  
Протокол від «31» серпня 2020 року № 1

Голова методичної комісії факультету комп'ютерних наук

  
Анатолій БЕРДНІКОВ

## ВСТУП

Програма навчальної дисципліни "Теорія алгоритмів" складена відповідно до освітньо-професійної програми підготовки першого (бакалаврського) рівня вищої освіти, спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології».

### 1. Опис навчальної дисципліни

1.1. Метою викладання навчальної дисципліни є засвоєння студентами знань та навичок по використанню та складанню алгоритмів, спеціалізованих для вирішення різних класів задач.

1.2. Основними завданнями вивчення дисципліни є:

- формування та обробка складних структур даних на основі базових типів;
- ознайомлення з ефективними методами зберігання інформації;
- розгляд великого класу алгоритмів для сортування даних;
- ознайомлення з особливостями розв'язання задач із графами;
- вивчення методів складання алгоритмів та пошуку найбільш ефективного для вирішення того чи іншого типу задач.

*Інтегральна компетентність.*

Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми, що характеризуються комплексністю та невизначеністю умов, під час професійної діяльності у галузі автоматизації або у процесі навчання, що передбачає застосування теорій та методів галузі.

*Загальні компетентності:*

- здатність застосовувати знання у практичних ситуаціях (K01);

*Спеціальні (фахові, предметні) компетентності (ФК)*

- Знання організації інформаційно-обчислювальних процесів і систем КЗП.03

1.3. Кількість кредитів – 4

1.4. Загальна кількість годин – 120

1.5. Характеристика навчальної дисципліни	
Нормативна / за вибором	
денна форма навчання	Заочна (дистанційна) форма навчання
Рік підготовки	
2-й	-й
Семестр	
3-й	-й
Лекції	
32 год.	_____ год.
Практичні, семінарські заняття	
_____ год.	_____ год.
Лабораторні заняття	
32 год.	_____ год.
Самостійна робота	
56 год.	_____ год.
У тому числі індивідуальні завдання	
40 год.	_____ год.

## 1.6. Заплановані результати навчання

### знати:

- алгоритми складання словників та кодування тексту;
- ефективні алгоритми пошуку інформації у масивах даних;
- алгоритми сортування;
- алгоритми вирішення завдань, пов'язаних із графами;
- методи розробки алгоритмів.

### вміти:

- застосовувати складні структури даних, методи та алгоритми для обробки різномірної інформації та вирішування складних обчислювальних завдань.

*В результаті вивчення дисципліни у студента повинні формуватися наступні програмні результати навчання (ПРН).*

- вміти застосовувати сучасні інформаційні технології та мати навички розробляти алгоритми та комп'ютерні програми з використанням мов високого рівня та технологій об'єктно-орієнтованого програмування, створювати бази даних та використовувати інтернет-ресурси. ПР03.
- вміти застосовувати методи системного аналізу, моделювання, ідентифікації та числові методи для розроблення математичних та імітаційних моделей окремих елементів та систем автоматизації в цілому, для аналізу якості їх функціонування із використанням новітніх комп'ютерних технологій. ПР06.

## 2. Тематичний план навчальної дисципліни

*Розділ 1. Складні структури даних.*

*Тема 1. Коди Хаффмана.*

Аналіз текстів та складання кодів кожного символу за алгоритмом Хаффмана. Повторення вмінь по використанню складених структур даних (запис, дерево, динамічний масив дерев) та роботі з текстовими файлами.

*Тема 2. Навантажене дерево.*

Складання словника за даним набором слів. Повторення та закріплення вмінь по використанню складених структур даних (запис, дерево, односпрямований список) та роботі з текстовими файлами.

*Тема 3. Хешування.*

Зберігання інформації з швидким пошуком. Методи розв'язання колізій за допомогою закритого та відкритого хешування та методу ланцюгів (масив односпрямованих списків).

*Тема 4. Піраміда.*

Принцип черги з пріоритетом на базі алгоритму піраміди.

*Розділ 2. Алгоритми сортування*

*Тема 5. Поняття сортування. Елементарні методи сортування. Швидке сортування.*

Сортування масиву простими методами (вибором, вставками, бульбашковим сортуванням). Повторення рекурсивного методу для швидкого сортування.

*Тема 6. Піромідалне сортування. Сортування злиттям.*

Використання алгоритму піраміди для сортування масиву. Вирішення проблеми сортування великого масиву, складеного з двох відсортованих, за допомогою алгоритму злиття.

*Тема 7. Кишенькове сортування. Сортування підрахунком. Зовнішнє сортування.*

Інші методи сортування. Вирішення проблеми сортування масиву, що за розмірами перевищує об'єм динамічної пам'яті (зовнішнє сортування у файлах).

### Розділ 3. Алгоритми на графах

#### Тема 8. Орієнтований, неорієнтований граф. Зважений граф.

Способи запису орієнтованих та неорієнтованих графів у комп'ютерній програмі. Поняття ваги ребра графа.

#### Тема 9. Пошук найкоротшого шляху в орієнтованому графі.

Алгоритми Дейкстри та Беллмана-Форда пошуку найкоротших шляхів від однієї вершини графа до інших.

#### Тема 10. Остовні дерева мінімальної вартості.

Алгоритми Крускала та Пріма для пошуку остовних дерев мінімальної вартості в неорієнтованому графі.

### Розділ 4. Методи розробки алгоритмів

#### Тема 11. Динамічне програмування.

Вирішення великої задачі шляхом виконання часткових завдань та зберігання відповідей.

#### Тема 12. "Жадібні" алгоритми.

Приклади "жадібних" алгоритмів з попередніх тем. Задача комівояжера.

#### Тема 13. Пошук з поверненням.

Пошук часткових рішень у задачах з перестановками.

## 3. Структура навчальної дисципліни

Назви розділів і тем	Кількість годин					
	Денна форма					
	Усього	у тому числі				
л		п	лаб	інд	сп	
1	2	3	4	5	6	7
<b>Розділ 1. Складні структури даних</b>						
Тема 1. Коди Хаффмана.	7	2		2		3
Тема 2. Навантажене дерево.	7	2		2		3
Тема 3. Хешування.	7	2		2		3
Тема 4. Піраміда.	7	2		2		3
<b>Разом за розділом 1</b>	<b>28</b>	<b>8</b>		<b>8</b>		<b>12</b>
<b>Розділ 2. Алгоритми сортування</b>						
Тема 5. Поняття сортування. Елементарні методи сортування. Швидке сортування.	13	4		4		5
Тема 6. Пірамідаліне сортування. Сортування злиттям.	14	4		4		6
Тема 7. Кишенькове сортування. Сортування підрахунком. Зовнішнє сортування.	13	4		4		5
<b>Разом за розділом 2</b>	<b>40</b>	<b>12</b>		<b>12</b>		<b>16</b>
<b>Розділ 3. Алгоритми на графах</b>						
Тема 8. Орієнтований, неорієнтований граф. Зважений граф.	11	4		4		3
Тема 9. Пошук найкоротшого шляху в орієнтованому графі.	7	2		2		3
Тема 10. Остовні дерева мінімальної вартості.	7	2		2		3
<b>Разом за розділом 3</b>	<b>25</b>	<b>8</b>		<b>8</b>		<b>9</b>
<b>Розділ 4. Методи розробки алгоритмів</b>						
Тема 11. Динамічне програмування.	5	1		1		3
Тема 12. "Жадібні" алгоритми.	5	1		1		3
Тема 13. Пошук з поверненням.	7	2		2		3
<b>Разом за розділом 4</b>	<b>17</b>	<b>4</b>		<b>4</b>		<b>9</b>
<b>Курсова робота</b>	<b>40</b>				<b>40</b>	
<b>Усього годин</b>	<b>150</b>	<b>32</b>		<b>32</b>	<b>40</b>	<b>46</b>

#### 4. Теми практичних (лабораторних) занять

№ з/п	Назва теми	Кількість годин
<b>Розділ 1. Складні структури даних</b>		
1	Коди Хаффмана.	2
2	Навантажене дерево.	2
3	Хешування.	2
4	Піраміда.	2
<b>Розділ 2. Алгоритми сортування</b>		
1	Елементарні методи сортування: вибором, вставками, бульбашкове.	4
2	Пірамідальне сортування. Сортування злиттям.	4
3	Кишенькове сортування. Сортування підрахунком.	4
<b>Розділ 3. Алгоритми на графах</b>		
1	Орієнтований та неорієнтований зважені графи.	4
2	Пошук найкоротшого шляху в графі (алгоритм Беллмана-Форда).	2
3	Остовні дерева мінімальної вартості (алгоритм Прима).	2
<b>Розділ 4. Методи розробки алгоритмів</b>		
1	Динамічне програмування.	1
2	"Жадібні" алгоритми.	1
3	Пошук з поверненням.	2
	<b>Разом</b>	<b>32</b>

#### 5. Завдання для самостійної роботи

№ з/п	Види, зміст самостійної роботи	Кількість годин
<b>Розділ 1. Складні структури даних</b>		
1	Для заданого тексту розробити словник на основі алгоритму навантаженого дерева. Реалізувати функцію перевірки тексту на орфографію. Підготовка до контрольної роботи	12
<b>Розділ 2. Алгоритми сортування</b>		
1	Розробити програму з використанням внутрішніх алгоритмів сортування (вибіркове, вставкою, "бульбашкове", швидке). Порівняти їх ефективності за швидкістю сортування для різних вихідних даних.	5
2	Реалізувати алгоритм зовнішнього сортування на основі алгоритму злиття. Відсортувати значення у заданому типізованому файлі з використанням додаткової пам'яті та без неї.	11
<b>Розділ 3. Алгоритми на графах</b>		
1	Роздрукувати найкоротші шляхи в орієнтованому графі з однієї вершини до всіх інших за допомогою алгоритму Дейкстри.	9
<b>Розділ 4. Методи розробки алгоритмів</b>		
1	Використати метод пошуку з поверненням для вирішення задачі розташування ферзів на дошці заданого розміру.	9
	Курсова робота	40
	<b>Разом</b>	<b>86 (56)</b>

## 6. Індивідуальні завдання (курсова робота)

Як наукове та навчально-дослідне завдання кожен студент має виконати курсову роботу, яка передбачає розробку комп'ютерної програми, що реалізує алгоритм, обраний студентом із списку запропонованих:

1. *Хешування.*  
Реалізувати алгоритми відкритого, закритого хешування та хешування методом ланцюжків.
2. *2-3-дерево.*  
Реалізувати основні операції для 2-3-дерева: додавання, видалення, друк, пошук значення.
3. *Червоно-чорне дерево.*  
Реалізувати основні операції для червоно-чорного дерева: додавання, видалення, друк, пошук значення.
4. *AVL-дерево.*  
Реалізувати основні операції для AVL-дерева: додавання, видалення, друк, пошук значення, балансування.
5. *Сильно пов'язані компоненти.*  
В орієнтованому графі знайти всі пов'язані компоненти та їх кількість.
6. *Порядкові статистики.*  
В заданому масиві шукати заданий за номером максимальний та заданий мінімальний елементи.
7. *Фільтр Блума.*  
Реалізувати Фільтр Блума з декількома хеш-функціями. Проаналізувати роботу алгоритма для різних параметрів (розміру фільтра, кількості елементів, кількості хеш-функцій).
8. *Дераміда.*  
Реалізувати основні операції для дераміди: додавання, видалення, друк, пошук значення, пошук пріоритету заданого значення.
9. *Алгоритм множення матриць Штрассена.*  
Реалізувати алгоритми множення матриць звичайним способом та способом Штрассена. Порівняти алгоритми за швидкістю для матриць рідного розміру.
10. *Тест Міллера-Рабіна.*  
Для заданого числа визначити, чи складне воно, чи просте.
11. *Алгоритм Кнута-Морріса-Пратта.*  
У заданому рядку шукати всі входження іншого заданого рядку. Таким чином проаналізувати текстовий файл.
12. *Тренажер «Прості схеми сортування».*  
Розробити програму, яка може навчити простим схемам сортування заданого масиву: вибіркового, вставкою, "бульбашковому". Передбачити наявність зрозумілого інтерфейсу.
13. *Тренажер «Швидке сортування».*  
Розробити програму, яка може навчити алгоритму швидкого сортування заданого масиву. Передбачити наявність зрозумілого інтерфейсу.
14. *Тренажер «Пірамідальне сортування».*  
Розробити програму, яка може навчити алгоритму пірамідального сортування заданого масиву. Передбачити наявність зрозумілого інтерфейсу.
15. *Тренажер «Сортування злиттям».*  
Розробити програму, яка може навчити алгоритму сортування заданого масиву злиттям. Передбачити наявність зрозумілого інтерфейсу.
16. *Алгоритм швидкого сортування та його візуалізація.*

- Реалізувати покрокове сортування заданого масиву методом швидкого сортування з візуалізацією основних етапів.
17. *Алгоритм пірамідального сортування та його візуалізація.*  
Реалізувати покрокове сортування заданого масиву методом пірамідального сортування з візуалізацією основних етапів.
  18. *Піраміда. Візуалізація основних операцій.*  
Реалізувати покрокову візуалізацію операцій додавання та видалення значень в піраміді.
  19. *Навантажене дерево. Візуалізація основних операцій.*  
Реалізувати покрокову візуалізацію операцій додавання, видалення та пошуку слів у словнику, який створено на основі алгоритму навантаженого дерева.
  20. *Алгоритм побудови кодів Хаффмана та його візуалізація.*  
Реалізувати покрокову візуалізацію пошуку бінарних кодів для кожного символу з наданого рядка за допомогою алгоритму Хаффмана.
  21. *2-3-дерево. Візуалізація основних операцій.*  
Реалізувати покрокову візуалізацію операцій додавання, видалення та пошуку значень у 2-3-дереві.
  22. *Червоно-чорне дерево. Візуалізація основних операцій.*  
Реалізувати покрокову візуалізацію операцій додавання, видалення та пошуку значень у червоно-чорному дереві.
  23. *АВЛ-дерево. Візуалізація основних операцій.*  
Реалізувати покрокову візуалізацію операцій додавання, видалення та пошуку значень в АВЛ-дереві.
  24. *Дераміда. Візуалізація основних операцій.*  
Реалізувати покрокову візуалізацію операцій додавання, видалення та пошуку значень у дераміді.
  25. *Неорієнтований граф. Візуалізація обходів в ширину та в глибину.*  
Для заданого неорієнтованого графа реалізувати покрокову візуалізацію алгоритму обходів в ширину та в глибину.
  26. *Візуалізація алгоритму Дейкстри для пошуку найкоротшого шляху в графі.*  
Для заданого орієнтованого зваженого графа реалізувати покрокову візуалізацію алгоритму Дейкстри для пошуку найкоротшого шляху з однієї вершини до всіх інших.
  27. *Візуалізація алгоритму Беллмана-Форда для пошуку найкоротшого шляху в графі.*  
Для заданого орієнтованого зваженого графа реалізувати покрокову візуалізацію алгоритму Беллмана-Форда для пошуку найкоротшого шляху з однієї вершини до всіх інших.
  28. *Візуалізація алгоритму Прима для побудови остовного дерева мінімальної вартості.*  
Для заданого орієнтованого зваженого графа реалізувати покрокову візуалізацію алгоритму Прима для побудови остовного дерева мінімальної вартості.
  29. *Візуалізація алгоритму Крускала для побудови остовного дерева мінімальної вартості.*  
Для заданого орієнтованого зваженого графа реалізувати покрокову візуалізацію алгоритму Крускала для побудови остовного дерева мінімальної вартості.

## 7. Методи навчання

На досягнення освітніх цілей спрямовані такі методи навчання студентів:

– *практичні* (використовують для пізнання дійсності, формування навичок і вмінь, поглиблення знань. Під час їх застосування використовуються такі прийоми: планування виконання завдання, постановка завдання, оперативне стимулювання, контроль і регулювання, аналіз результатів, визначення причин недоліків);



– *пояснювальне-ілюстративний* (використовують для викладання й засвоєння нового навчального матеріалу, фактів, підходів, оцінок, висновків тощо);

– *репродуктивний* (для застосування студентами вивченого на основі зразка або правила, алгоритму, що відповідає інструкціям, правилам, в аналогічних до представленого зразка ситуаціях);

Як правило лекційні та практичні заняття проводяться аудиторне. В умовах дії карантину заняття проводяться відповідно до Наказу ректора Харківського національного університету імені В.Н. Каразіна (аудиторне або дистанційно за допомогою платформ Google Meet або Zoom).

## 8. Методи контролю

Контроль засвоєння навчального матеріалу здійснюється шляхом:

- прийому та оцінювання звітів з виконання індивідуальних завдань з курсової роботи;
- прийому та оцінювання завдань самостійної роботи;
- проведення контрольної роботи;
- проведення письмового підсумкового контролю знань.

## 9. Схема нарахування балів

Поточний контроль та самостійна робота													Контрольна робота, передбачена ПП	Індивідуаль не завдання (КП)	Разом	Залік	Сума
Розділ 1				Розділ 2			Розділ 3			Розділ 4							
T1	T2	T3	T4	T1	T2	T3	T1	T2	T3	T1	T2	T3					
2	2	2	2	2	2	2	2	4	4	2	2	2	10	20	60	40	100

### Критерії оцінювання практичних занять

Протягом семестру студент має виконати 15 практичних завдань за відповідними темами. Кожне завдання складається з двох пунктів (алгоритмів або їх фрагментів) і оцінюється в 2 бали: або студент їх виконав коректно, або ні. Таким чином, загалом за практичні завдання можна отримати 30 балів.

### Критерії оцінювання контрольної роботи

Студент має виконати контрольну роботи з курсу, яка передбачена навчальним планом підготовки. Робота оцінюється в 10 балів.

#### 9-10 балів:

- студент демонструє глибоке розуміння теми питання
- студент повністю розкриває сутність питання
- в роботі наведені приклади, якщо це необхідно
- текст роботи викладено лаконічно, чітко, логічно та послідовно.
- робота демонструє високий рівень засвоєння матеріалу курсу

#### 7-8 балів:

- студент демонструє розуміння теми питання
- студент в цілому розкриває сутність питання
- в роботі наведені деякі з необхідних прикладів
- текст роботи викладено в цілому логічно та послідовно
- робота демонструє хороший рівень засвоєння матеріалу курсу

**5-6 балів:**

- студент демонструє базове розуміння теми питання
- студент частково розкриває сутність питання
- в роботі не наведені необхідні приклади
- текст роботи викладено в цілому логічно, але не завжди послідовно
- робота демонструє середній рівень засвоєння матеріалу курсу

**3-4 балів:**

- студент демонструє часткове розуміння теми питання
- студент не достатньо розкриває сутність питання
- в роботі не наведені необхідні приклади
- текст роботи викладено дещо хаотично та не завжди логічно
- робота демонструє часткове засвоєння матеріалу курсу

**1-2 балів:**

- студент дещо помилково розуміє тему питання
- студент не розкриває сутність питання
- в роботі не наведені необхідні приклади
- текст роботи викладено хаотично та не логічно
- робота демонструє мінімальний рівень засвоєння матеріалу курсу

**0 балів: робота відсутня****Критерії оцінювання курсової роботи**

Курсова робота оцінюється сумарно в 20 балів. Оцінка складається з таких елементів:

<b>Критерій</b>	<b>Бали</b>
Документація та теоретична частина	5
Проектування і опис програми	5
Код у відповідності з вимогами	4
Тестування	2
Якість презентації	2
Доповідь та відповіді на запитання	2
<b>Всього</b>	<b>20</b>

**Критерії оцінювання виконання залікової письмової роботи**

В кінці семестру студент має виконати письмову залікову роботу. Ця робота виконуються в аудиторії. Вона розрахована на 1 пару (80 хвилин), складається з двох питань, на які треба дати вільну розгорнуту відповідь, та оцінюються у 40 балів. Кожне питання має фіксовану максимальну кількість балів 20 та оцінюється окремо таким чином:

- повна, розгорнута відповідь, що містить приклади та додаткові відомості, які були опрацьовані протягом семестру на самостійній роботі – 100 % від запланованої кількості балів за питання;
- повна, але не розгорнута відповідь, відсутність достатньої кількості прикладів – 90 % від запланованої кількості балів за питання;

- повна, але не розгорнута відповідь, що містить незначні помилки чи неточності – 80 % від запланованої кількості балів за питання;
- неповна відповідь, що не містить критичних помилок та неточностей – 70 % від запланованої кількості балів за питання;
- неповна відповідь, що містить істотні, але не критичні помилки або неточностей – 50 % від запланованої кількості балів за питання;
- відповідь, що в цілому вірна, але містить критичну помилку чи неточність – 25 % від запланованої кількості балів за питання;
- відповідь не вірна, не дана на поставлене запитання або відсутність відповіді оцінюється в 0 балів.

### Шкала оцінювання

Сума балів за всі види навчальної діяльності протягом семестру	Оцінка	
	для чотирирівневої шкали оцінювання	для дворівневої шкали оцінювання
90 – 100	відмінно	зараховано
70-89	добре	
50-69	задовільно	
1-49	незадовільно	не зараховано

## 10. Рекомендована література

### Основна література

1. Кормен Т. Алгоритмы: построение и анализ. – М.: Издательский дом "Вильямс", 2013. – 1296 с.
2. Ахо А., Хопкрофт Дж., Ульман Дж. Структуры данных и алгоритмы. – М.: Издательский дом "Вильямс", 2003. – 384 с.
3. Дасгупта С., Пападимитриу Х., Вазирани У. Алгоритмы. – М.: МЦНМО, 2014. – 320 с.
4. Вирт Н. Алгоритмы и структуры данных. – М.: Мир, 1989.
5. Кнут Д. Искусство программирования. Т.1. Основные алгоритмы. – М.: Издательский дом "Вильямс". – 2000.
6. Кнут Д. Искусство программирования. Т.3. Поиск и сортировка. – М.: Издательский дом "Вильямс". – 2000.

### Допоміжна література

1. Дейкстра Э. Дисциплина программирования. – М.: Мир, 1978.
2. Беллман Р. Динамическое программирование. – М., ИЛ., 1960.
3. Берж С. Теория графов и ее применение. – М., ИЛ., 1962.
4. Грин Д., Кнут Д. Математические методы анализа алгоритмов. – М.: Мир, 1987.

### 11. Посилання на інформаційні ресурси в Інтернеті, відео-лекції, інше методичне забезпечення

1. Алгоритмы сортировки: [Електроний ресурс] // Алгоритмы методы исходники, [Илья Кантор](http://algolist.manual.ru/sort), 2000-2017. Режим доступа: <http://algolist.manual.ru/sort>.
2. Алгоритмы на графах: [Електронний ресурс] // Wikipedia the free encyclopedia, 2001-2017. Режим доступа: [https://uk.wikipedia.org/wiki/%D0%9A%D0%B0%D1%82%D0%B5%D0%B3%D0%BE%D1%80%D1%96%D1%8F:%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D0%B8\\_%D0%BD%D0%B0\\_%D0%B3%D1%80%D0%B0%D1%84%D0%B0%D1%85](https://uk.wikipedia.org/wiki/%D0%9A%D0%B0%D1%82%D0%B5%D0%B3%D0%BE%D1%80%D1%96%D1%8F:%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D0%B8_%D0%BD%D0%B0_%D0%B3%D1%80%D0%B0%D1%84%D0%B0%D1%85).