

Скачати з DL архів із заглушкою проекту. Архів розпакувати та імпортувати вміст у IDE (Eclipse / IDEA) як Maven project. Відповідно до завдання внести зміни до проекту. Досягнути проходження тестів. Завантажити рішення на DL.

Зауваження.

1. Кореневий пакет: ua.nure.jfn.task3

2. У завданні 1-3 у вхідній інформації може бути:

- латиниця (символи алфавіту англійської мови, інші символи ASCII залежно від завдання, докладно див. тести);
- кирилиця (символи алфавіту української мови).

3. Якщо програма зчитує інформацію з файлу, необхідно вказати кодування, в якому записана інформація. Використовувати кодування CP1251.

Розділювач рядків у файлах `\r\n` або `\n`.

Див. метод `Utils.getContent`.

4. Кожен клас `PartX`, та кожен конвертер повинен мати метод `main`, який демонструє роботу відповідного підзавдання. У кореновому пакеті створити клас `Demo`, який демонструє роботу всього написаного функціоналу.

5. Заборонено використовувати:

- 1) контейнерні класи
- 2) типи з пакета `java.util`
- 3) типи з пакета `java.util.stream`
- 4) класи оболонки.

6. Всі роботи будуть перевірені на плагіат. Якщо буде детектовано плагіат, оцінку буде знижено.

7. Робота буде оцінена в діапазоні $[0..100*k]$, де k залежить від форми здачі л.р.

Варіанти:

(1) Залити архів із рішенням на DL: $k=0.7$

Максимальна оцінка – 70 балів.

(2) Залити архів із рішенням на DL, а також залити текстовий файл з назвою `link.txt` (кодування файлу Unicode у форматі UTF-8), в якому розмістити посилання на відеозапис з поясненням, як було вирішено завдання: $k=0.9$

Максимальна оцінка – 90 балів.

(3) залити архів із рішенням на DL, захистити л.р. шляхом спілкування з викладачем: $k=1$

Максимальна оцінка – 100 балів.

Завдання 1

Назва класу: ua.nure.jfn.task3.Part1

Вхідну інформацію завантажувати з файлу part1.txt

Завдання має бути вирішене за допомогою регулярних виразів.

Створити клас зі статичними методами, які перетворюють вхідну інформацію на вихідну.

Вхідна інформація має наступну структуру (значення Login/Name/Email в загальному випадку можуть бути будь-якими;

Login і Name можуть містити латиницю та кирилицю):

Input data (part1.txt)

```
Login;Name;Email
petrenko;Марія Петренко;petrenko2@mail.com
ivanenko;Беатрисса-Валері Іваненко;ivanenko@mail.com
obama;Barack Hussein Obama;obama@gmail.com
lennon;John Lennon;lennon@live.com
bush;Джордж Вокер Буш;bush@gmail.com
```

Методи, які потрібно написати, мають такий вигляд (N – цифра: 1, 2, 3):

```
public static String convertN(String input) {
    ...
}
```

1.1. Метод convert1

Повинен перетворювати input data на рядок наступного виду:

Output of convert1

```
LastName;Email
Петренко;petrenko2@mail.com
Іваненко;ivanenko@mail.com
Obama;obama@gmail.com
Lennon;lennon@live.com
Буш;bush@gmail.com
```

1.2. Метод convert2

Повинен перетворювати input data на рядок наступного виду:

Output of convert2

```
LastName;MiddleName;FirstName
Петренко;;Марія
Іваненко;;Беатрисса-Валері
Obama;Hussein;Barack
Lennon;;John
Буш;Вокер;Джордж
```

1.3. Метод convert3

Повинен перетворювати input data в рядок наступного виду:

Output of convert3

(використовуйте моноширинний шрифт, наприклад Consolas, щоб побачити вирівнювання, або див. файл part1_convert3_output.png)

```
-----
Петренко Марія          petrenko
Іваненко Беатрисса-Валері ivanenko
Обама Barack Hussein   obama
Леннон John             lennon
Буш Джордж Вокер       bush
-----
```

Завдання 2

Назва класу: ua.nure.jfn.task3.Part2

Вхідну інформацію завантажувати з файлу part2.txt

Даний текст: рядки, в яких записані слова через пробіл.
Словом вважати послідовність непробельних символів.

Всі слова в тексті можуть бути впорядковані таким чином:

спочатку йдуть слова з довжиною L1

потім йдуть слова з довжиною L2

...

в останню чергу йдуть слова з довжиною Ln

Числа $L1 < L2 < \dots < Ln$ утворюють висхідну послідовність натуральних чисел, наприклад: $1 < 3 < 5 < 6 < 7$.

Написати метод convert, який приймає текст у вигляді рядка та натуральне число k та повертає:

(a) об'єднані пробілом слова, які мають довжину Lk, якщо $1 \leq k \leq n$;

(b) порожній рядок, якщо $k > n$.

Результат не повинен включати дублікати слів.

Формат виведення дивись на прикладах нижче

Заглушка методу

```
-----
public static String convert(String input, int k) {
    ...
}
-----
```

Приклад Input data (part2.txt)

When I was younger, so much younger than today
I never needed anybody's help in any way

Output (k=1)

1: I

Output (k=2)

2: so in

Output (k=8)

8: younger,

Output (k=9)

8: anybody's

Завдання 3

Назва класу: ua.nure.jfn.task3.Part3

Вхідну інформацію завантажувати з файлу part3.txt

Завдання має бути вирішене за допомогою регулярних виразів.

Створити статичний метод convert, який перетворює вхід у вихід.

Заглушка методу

public static String convert(String input) {
 ...
}

Вхід: рядки, в яких записані слова через пробіл.

Словом вважати послідовність, що містить лише літери (всі інші символи до складу слова не входять).

Вихід: вхідний текст, але регістр кожного парного за рахунком дубліката слова інвертовано.

Приклад Input data

when I was younger younger than today was
I never needed needed never never was I was

Output

when I was younger YOUNGER than today WAS
i never needed NEEDED NEVER never was I WAS

Завдання 4

Назва класу: ua.nure.jfn.task3.converter.CyrillicConverter

Реалізувати метод

```
public static String convert(String str) {  
    ...  
}
```

який конвертує з десяткової системи числення до кириличної системи числення.
Див. https://en.wikipedia.org/wiki/Cyrillic_numerals

Кириличні цифри та їх значення у десятковій системі числення дано у заглушці.
Вхідний рядок може містити символ '_' для розділення розрядів.
Область визначення методу: [1..999_999]

Приклад Input data

999_999

Output (файл converters_output.png):

ѳцѳчѳѳцѳѳ

Завдання 5

Назва класу: ua.nure.jfn.task3.converter.MayaConverter

Реалізувати метод

```
public static String convert(String str) {  
    ...  
}
```

який конвертує з десяткової системи числення до системи числення мая.
Див. https://en.wikipedia.org/wiki/Maya_numerals

Цифри мая та їх значення у десятковій системі числення дано у заглушці.
Вхідний рядок може містити символ '_' для розділення розрядів.
Область визначення методу: [1..Integer.MAX_VALUE]

Приклад Input data

100_000

Output (файл converters_output.png):

• = III = 0

Примітка: для відображення символів мая у консолі IDE ви можете використовувати шрифт
'Noto Sans Mayan Numerals' або 'BabelStone Mayan Numerals'

Завдання 6

Назва класу: ua.nure.jfn.task3.converter.PositionalConverter

Реалізувати метод

```
public static String convert(String str) {  
    ...  
}
```

який конвертує число number з системи числення по основанню baseFrom в систему счислення по основанню baseTo.

Вхідна інформація має такий формат: baseFrom:number:baseTo

Число number може містити символ '_' для розділення розрядів.
Область визначення методу: [1..Integer.MAX_VALUE]

baseTo и baseFrom можуть бути з діапазону [2..36].

Система числення за основою 36 використовує такі цифри (у дужках дано десяткове значення):

0 (=0)

1 (=1)

...

9 (=9)

A (=10)

B (=11)

...

Z (=35)

Приклад Input data

10:17:2

Output

10001

Завдання 7

Назва класу: ua.nure.jfn.task3.converter.RomanConverter

Реалізувати метод

```
public static String convert(String str) {  
    ...  
}
```

який конвертує з десяткової системи числення до римської системи числення.
Див. https://en.wikipedia.org/wiki/Roman_numerals

Римські цифри та їх значення у десятковій системі числення дано у заглушці.
Вхідний рядок може містити символ '_' для розділення розрядів.
Область визначення методу: [1..3_999]

Приклад Input data

3_999

Output (файл converters_output.png):

MMMCMXCIX

Примітка: для відображення символів римської системи числення у консолі IDE ви можете будь-який з наступних шрифтів:

'Noto Sans Mayan Numerals', 'BabelStone Mayan Numerals', 'Consolas'.

Запитання.

1. Що таке кодування, приклади кодувань, кодування за замовчуванням.
2. Unicode. UTF. Види, відмінності. BE/LE варіанти. BOM.
3. Рядкові класи, відмінності.
4. Пул рядків, навіщо потрібен. Як працює метод intern.
5. Регулярні вирази. Навіщо потрібні? Написати приклад.
6. Навіщо призначені класи Pattern/Matcher?
7. Які методи класу String працюють з регулярними виразами?
8. i18n/l10n. В чому різниця. Які засоби є для роботи з i18n?
9. З яких символів складається символ нового рядка?
10. Чим жадібний регулярний вираз відрізняється від лінивого?