**Algorytm zachłanny – wydawanie reszty, liczby rzymskie**

**Przykład 1**

Mając do dyspozycji nominały 200, 100, 50, 20, 10, 5, 2, 1 wydaj resztę 393 używając jak najmniejszą liczbę nominałów.

393 = 1\*200+1\*100+1\*50+2\*20+0\*10+0\*5+1\*2+1\*1

**Specyfikacja algorytmu:**

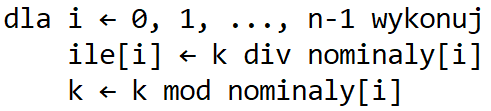
**Dane:**

n – liczba dostępnych nominałów  
nominaly[0..n-1] – ciąg liczb oznaczających dostępne nominały posortowane malejąco  
k – kwota, która należy wydać za pomocą nominałów

**Wyniki:**

ile[0..n-1] – ciąg liczb, w którym ile[i] oznacza ile nominałów nominaly[i] należy użyć, aby wydać resztę w kwocie k

Pseudokod



**Zadanie 1**

Napisz program w języku Python, który wczyta nominały z pliku nominaly.txt (są to liczby w pierwszym wierszu pliku rozdzielone spacjami), zapyta użytkownika o podanie kwoty reszty do wydania, a następnie wyświetli sposób wydania reszty za pomocą jak najmniejszej liczby nominałów metodą zachłanną w formacie:

Np. dla kwoty 140 wynikiem działania programu będzie:

1x100  
2x20

**Informacje wstępne do zadań 2-5**

Zapoznaj się z opisem rzymskiego systemu zapisywania liczb:

<https://pl.wikipedia.org/wiki/Rzymski_system_zapisywania_liczb>

**Zadanie 2.**

Zamień podane liczby w systemie rzymskim na liczby w systemie dziesiętnym:

1. MDCCLXXXV
2. MMCDXCIV
3. MCXCIII
4. MMMCXXIII

**Zadanie 3.**

Zamień liczby podane w systemie dziesiętnym na liczby w systemie rzymskim:

1. 1293
2. 2189
3. 3743
4. 999

**Zadanie 4.**

Napisz program realizujący zamianę liczb z przedziału [1, 4000) z systemu dziesiętnego na rzymski, wykorzystując algorytm zachłanny.

**Wskazówki:**

Przyjmij jako nominały ciąg 1000, 900, 500, 400, 100, 90, 50, 40, 10, 9, 5, 4, 1 oraz odpowiadające im wartości M, CM, D, CD, C, XC, L, XL, X, IX, V, IV, I

Zamiast wykonywania dzielenia z resztą można wielokrotnie odejmować wartość, dopóki jest to możliwe.

**Zadanie 5.**

Napisz program realizujący zamianę liczb z systemu rzymskiego na dziesiętny. Uwzględnij możliwość użycia symboli M, D, C, L, X, V, I.

**Wskazówki:**

Gdy następny symbol ma większą wartość, to wartość aktualnego symbolu jest odejmowana.

Do przechowywania wartości symboli można wykorzystać strukturę słownika (ang. dictionary)

**Przykład 2**

Program, który prosi użytkownika liczby rzymskiej o długości 1 i wyświetla wartość tej liczby w systemie dziesiętnym.

