

Politechnika Śląska w Gliwicach  
Wydział Automatyki, Elektroniki i Informatyki



# Podstawy Programowania Komputerów

Konwerter

---

autor	Dawid Kaczor
prowadzący	dr inż. Krzysztof Simiński
rok akademicki	2016/2017
kierunek	informatyka
rodzaj studiów	SSI
semestr	1
termin laboratorium	wtorek, 12:00-13:30
grupa	6
sekcja	2
termin oddania sprawozdania	2016-12-02
data oddania sprawozdania	2016-12-02

---

# 1 Treść zadania

Napisać program do konwersji liczb naturalnych z systemu o podstawie  $p \in [2, 36]$  na system o podstawie  $q \in [2, 36]$ . Liczby do konwersji podawane są w pliku wejściowym. Każda liczba podana jest w osobnym wierszu w następujący sposób:  $234 \ (6 \rightarrow 12)$ , co oznacza, że liczbę 234 należy przekształcić z systemu szóstkowego na dwunastkowy. Wyniki zapisywane są do pliku wyjściowego. Brakujące cyfry w systemach o podstawie  $n > 10$  uzupełniane są kolejnymi literami alfabetu łacińskiego rozpoczynając od a. Program jest uruchamiany z linii poleceń. Po przełączniku `-i` podawana jest nazwa pliku wejściowego, po `-o` nazwa pliku wyjściowego. Przełączniki mogą być podane w dowolnej kolejności.

## 2 Analiza zadania

Zagadnienie przedstawia problem konwersji liczb naturalnych zapisanych w pliku.

### 2.1 Algorytmy

Program powinien działać w czterech etapach. Pierwszym z nich jest wczytanie danych z pliku i selekcja danych liczbowych. Następnie liczbę należy przekonwertować na system dziesiętny po czym na system docelowy. Tak zamieniona liczba ma zostać zapisana w osobnej linii w pliku wyjściowym/

## 3 Specyfikacja zewnętrzna

Program jest uruchamiany z linii poleceń. Należy przekazać do programu nazwy plików: wejściowego i wyjściowego po odpowiednich przełącznikach (odpowiednio: `-i` dla pliku wejściowego i `-o` dla pliku wyjściowego), np.

```
program -i wejscie.txt -o wyjscie.txt
program -o wyjscie.txt -i wejscie.txt
```

Przełączniki mogą być podane w dowolnej kolejności. Uruchomienie programu bez żadnego parametru lub z parametrem `-h`

```
program
program -h
```

jak i uruchomienie programu z nieprawidłowymi parametrami powoduje wyświetlenie komunikatu

Witam w programie konwertującym liczby naturalne w różnych systemach liczbowych!

Jak uruchomić program?

W wierszu poleceń należy wpisać nazwę programu (konwerter.exe), następnie po przełączniku -i nazwę pliku tekstowego z danymi wejściowymi i po przełączniku -o nazwę pliku tekstowego w którym mają zapisane zostać dane wejściowe.

Jak mają wyglądać dane wejściowe?

a ( p -> q ), gdzie:

a - liczba do konwersji

p - system liczbowy (od 2 do 36), w którym zapisana jest liczba "a"

q - system docelowy

UWAGA! Dla systemów powyżej 10, należy wpisywać wielkie litery.

## 4 Specyfikacja wewnętrzna

W programie funkcje operujące na plikach i zmiennych zostały oddzielone od funkcji głównej.

### 4.1 Funkcje

```
bool odczyt (int ile, char ** argumenty, string
&szinput, string &szoutput)
```

Funkcja wczytuje parametry wywołania programu i sprawdza ich poprawność, są one podawane w linii poleceń jako przełączniki w dowolnej kolejności

```
-i input
-o output
-h
```

Funkcja pobiera następujące parametry

ile	liczba parametrów podanych przy uruchomieniu programu
argumenty	tablica wskaźników na łańcuchy użytych przy uruchomieniu
szinput	parametr wyjściowy, do którego zostanie zapisana odczytana nazwa pliku wejściowego
szoutput	parametr wyjściowy, do którego zostanie zapisana odczytana nazwa pliku wyjściowego

Funkcja zwraca następujące wartości

<code>true</code>	podano prawidłowe parametry wywołania programu, zostały one odczytane i zapisane do <code>szinput</code> i <code>szoutput</code>
<code>false</code>	nie udało się odczytać wartości parametrów wywołania programu, wartości zmiennych <code>szinput</code> i <code>szoutput</code> mogą nie zawierać istotnych wartości

Funkcja `DecToAll` konwertuje liczbę podaną w systemie dziesiętnym na dowolny system z zakresu od 2 do 36

```
string DecToAll(int liczba, int sys)
```

Funkcja przyjmuje następujące parametry

<code>liczba</code>	liczba naturalna w systemie dziesiętnym którą należy przekonwertować
<code>sys</code>	podstawa systemu, na który należy przekonwertować podaną liczbę

Funkcja zwraca wartość typu **string** zawierającą przekonwertowaną liczbę.

Funkcja `AllToDec` konwertuje liczbę podaną w dowolnym systemie z zakresu od 2 do 26 na liczbę w systemie dziesiętnym.

```
int AllToDec(const string &liczba, int sys)
```

Funkcja przyjmuje następujące parametry

<code>liczba</code>	liczba naturalna w dowolnym systemie
<code>sys</code>	podstawa systemu, w którym zapisana jest liczba

Funkcja zwraca liczbę naturalną w systemie dziesiętnym

Funkcja `uzs` zmienia w łańcuchu wszystkie znaki, które nie są literą ani cyfrą, na znak biały

```
string uzs(string &linia)
```

Funkcja przyjmuje następujące parametry

Linia                      łańcuch na którym należy dokonać zamiany

Funkcja zwraca łańcuch bez znaków niealfanumerycznych.

## 5 Testowanie

Program został przetestowany na różnego rodzaju danych wejściowych. Przy prawidłowych parametrach nie zaobserwowano żadnych błędnych zapisów. Błędnie wprowadzona linia (np. liczba przekraczająca zakres systemu) powoduje wypisanie informacji o błędnych danych. Pusta linia powoduje przejście do następnej linii w pliku wejściowym i wypisanie pustej linii w pliku wyjściowym.

## 6 Wnioski

Program konwertujący systemy liczbowe jest prostym programem jeśli użyje się odpowiednich algorytmów. Pomocna okazała się znajomość sposobu zamiany liczb pomiędzy systemem binarnym i dziesiętnym gdyż po dodaniu pewnych zmiennych można było otrzymać uniwersalny algorytm. Dużym problemem okazała się praca na plikach i łańcuchach. Tworzenie kontroli błędów wymagało wymyślenia kilku sposobów na podanie błędnych danych i stworzenie odpowiednich warunków. Projekt nauczył mnie lepszego podejścia proceduralnego i dostrzegania pozornie małych błędów które mają duży wpływ na działanie programu.