

zaawansowane techniki C++ i STL**wyrażenia regularne**

Instytut Informatyki
Uniwersytetu Wrocławskiego

Paweł Rzechonek

Zadanie 1 (1 pkt)

Zdefiniuj wyrażenie regularne do rozpoznawania prawidłowo zapisanej godziny. Zapis godziny to liczba godzin (0-23) i liczba minut (0-59) oddzielone dwukropkiem (obydwie liczby zapisane za pomocą dwóch cyfr); opcjonalnie w godzinie mogą się pojawić sekundy (0-59).

Przykłady prawidłowo zapisanych godzin: 14:17, 09:15, 23:37:08.

Przykłady wadliwie zapisanych godzin: 14.16 (kropka zamiast dwukropka), 19:5 (minuty zapisane jedną cyfrą), 21:32:07.10 (zbyt duża dokładność).

Przetestuj wyrażenie na danych wprowadzonych ze standardowego wejścia.

Zadanie 2 (2 pkt)

Zdefiniuj wyrażenie regularne do rozpoznawania prawidłowo zapisanej daty. Zapis daty składa się z numeru dnia w miesiącu (1-31), numeru miesiąca (1-12) i roku; liczby te mają być od siebie oddzielone minusem (numer dnia i miesiąca zapisane za pomocą dwóch cyfr a rok za pomocą czterech). Zadbaj o sprawdzanie górnych ograniczeń na liczbę dni w poszczególnych miesiącach (nie musisz sprawdzać przestępności roku, chociaż byłoby to ciekawe wyzwanie – można założyć, że luty ma 28 dni).

Przykłady prawidłowo zapisanych dat: 15-09-1999, 05-02-2018, 10-12-2025.

Przykłady wadliwie zapisanych dat: 03/07/1917 (ukośnik zamiast myślnika), 6-3-1945 (dni i miesiące zapisane jedną cyfrą), 29-02-2022 (liczba dni w miesiącu lutym przekracza 28).

Przetestuj wyrażenie na danych wprowadzonych ze standardowego wejścia.

Zadanie 3 (2pkt)

Zdefiniuj wyrażenie regularne do rozpoznawania prawidłowo zapisanej nazwy miejscowości w Polsce (nie używaj polskich znaków diakrytycznych, tylko podobnych liter w alfabecie angielskim, na przykład zamiast litery ż użyj litery z). Zapis nazwy miejscowości to ciąg słów, gdzie każde słowo rozpoczyna się z dużej litery. Poszczególne słowa mogą być odseparowane dowolną liczbą białych znaków. W nazwach miejscowości zdarzają się też połączenia słów za pomocą myślnika (w okolicach łącznika nie występują białe znaki).

Przykłady prawidłowo zapisanych nazw miejscowości: Wrocław, Bielsko-Biała, Tarnowskie Góry.

Przykłady wadliwie zapisanych nazw miejscowości: WARSZAWA (cała nazwa zapisana dużymi literami), Zle mieso (druga część nazwy z małej litery), Zimna-Wodka-07 (trzecia część nazwy zawiera cyfry).

Przetestuj wyrażenie na danych wprowadzonych ze standardowego wejścia.

Zadanie 4 (2 pkt)

Zdefiniuj wyrażenie regularne do rozpoznawania prawidłowo zapisanej liczby zespolonej. Zapis liczby zespolonej składa się z dwóch liczb rzeczywistych (z opcjonalną możliwością wystąpienia części ułamkowej po kropce dziesiętnej) połączonych znakiem plusa albo minusa (w pierwszej liczbie opcjonalny znak minusa), przy czym druga liczba jest zakończona małą albo dużą literą „i”; całość jest ujęta w nawiasy okrągłe.

Przykłady prawidłowo zapisanych liczb zespolonych: $(12+3i)$, $(7.4-0.5i)$, $(2+0.01i)$.

Przykłady wadliwie zapisanych liczb zespolonych: $3+4i$ (brak nawiasów okrągłych obejmujących liczbę), $(3,14-2,72i)$ (przecinek zamiast kropki dziesiętnej), $(5.7i)$ (brak części rzeczywistej).

Przetestuj wyrażenie na danych wprowadzonych ze standardowego wejścia.

Zadanie 5 (3 pkt)

Zdefiniuj wyrażenie regularne wykrywające hiperłącze w dokumencie HTML – chodzi o znacznik ` ... `. Następnie użyj tego wyrażenia do wypisania na standardowym wyjściu wszystkich znalezionych w tekście hiperłączy (bez opakowania w postaci znacznika `<a>` – sama treść atrybutu `href`).

Uwaga 1: atrybut `href` nie musi być pierwszym atrybutem w znaczniku `<a>`, na przykład:

```
<a class="kl" href="https://ii.uni.wroc.pl/">
```

Uwaga 2: znacznik `<a>` nie jest jedynym w HTML, który może posiadać atrybut `href`, na przykład:

```
<link rel="shortcut icon" href="logo.svg" />
```

Przetestuj wyrażenie regularne na tekście z pliku w formacie HTML (najczęściej z rozszerzeniem `.html` lub `.php`) wskazanego za pomocą argumentów wywołania programu.