

Założenia algorytmiczne

Najpierw zdefiniowane są stałe reprezentujące te częstości dla języków angielskiego i polskiego. Następnie użytkownik wprowadza tekst do analizy. Program oblicza, jak często występuje każda litera w tekście.

Po uzyskaniu tych danych porównuje uzyskane częstości z wcześniej zdefiniowanymi wartościami dla obu języków. Dla każdego języka oblicza odległość między rzeczywistą częstością a oczekiwaną. Im mniejsza różnica, tym bardziej prawdopodobne, że tekst jest napisany w danym języku.

Na koniec wskazuje szacowany język wprowadzonego tekstu. Algorytm opiera się na założeniu, że różnice w częstościach występowania liter mogą być wykorzystane do identyfikacji języka tekstu.

1. Funkcja **analyzer_chars_frequency**

- Funkcja przyjmuje listę znaków i tekst do analizy.
- Wszystkie litery w tekście są zamieniane na wielkie.
- Dla każdego znaku z listy, funkcja liczy ilość jego wystąpień w tekście.
- Obliczane są częstości występowania poszczególnych znaków.
- Funkcja zwraca słownik, gdzie klucze to znaki, a wartości to ich częstości występowania.

2. Funkcja **calculate_differences**

- Funkcja przyjmuje dwa słowniki reprezentujące częstości występowania znaków.
- Oblicza odległość euklidesową między częstościami znaków w obu słownikach.
- Zwraca obliczoną odległość.

3. Funkcja **print_frequencies**

- Funkcja pomocnicza do czytelnego wydruku częstości występowania znaków.
- Wydrukuję znaki i ich procentową częstość występowania, grupując po 5 znaków w jednym wierszu.

4. Funkcja **main**

- Pobiera tekst od użytkownika.
- Sprawdza, czy tekst nie jest dłuższy niż 100 znaków.
- Wywołuje funkcję **analyzer_chars_frequency** dla częstości angielskich i polskich.
- Za pomocą **calculate_differences** oblicza różnice w częstościach dla obu języków.
- Na podstawie różnic decyduje, który język jest bardziej prawdopodobny.
- Wydrukuję wynik za pomocą **print_frequencies**.