

Projekt indywidualny Specyfikacja Funkcjonalna

Arkadiusz Michalak

Algorytmy i struktury danych
07.11.2018

1 Opis problemu

Celem projektu jest stworzenie programu którego zadaniem będzie przeliczanie wymian walutowych. Będąc zainteresowany wymianą waluty z jednej na drugą można to zrobić bezpośrednio bądź też pośrednio przy użyciu innych walut niż wejściowa i docelowa. Określenie który ze sposobów będzie korzystniejszy i w jakiej kolejności wymieniać pośrednie waluty stwarza duże wyzwanie. Dzieje się tak ze względu na znaczną ilość różnych walut i rozmaite powiązania między nimi, oraz ewentualne prowizji od wymiany, zarówno stałe jak i wyrażone procentem kwoty którą otrzymujemy. Analizując zestaw danych można znaleźć sytuacje w której wymieniając jedną walutę wracamy do posiadania tej samej waluty jednak osiągamy zysk. Takie działanie będziemy określać mianem *arbitrażu ekonomicznego*.

2 Funkcjonalność

Program będzie posiadał dwie funkcjonalności:

- wskazanie najkorzystniejszego sposobu wymiany danej waluty wejściowej na określoną walutę wyjściową
- sprawdzenie czy w podanym zbiorze danych istnieje arbitraż ekonomiczny, następnie wypisanie go

3 Sposób obsługi

Do uruchomienia programu wymagane jest zainstalowanie maszyny wirtualnej java. Obsługa będzie odbywać się w całości z wiersza polecań, wywołując program należy podać odpowiednie argumenty w następującej kolejności:

1. którą funkcjonalność uruchomić:
 - arbitraz - w celu sprawdzenia czy istnieje arbitraz
 - wymiana - w celu znalezienia najbardziej opłacalnego sposobu wymian
2. ścieżka do pliku zawierającego dane, format i sposób organizacji danych w pliku opisany w sekcji **4 Sposób organizacji pliku**
3. nazwa waluty wejściowej, wyrażoną zunifikowanym 3-literowym kodem
4. ilość waluty wejściowej
5. nazwa waluty wyjściowej, jeśli wybrano opcje wymiany

4 Sposób organizacji pliku

Plik wejściowy musi być w formacie tekstowym *.txt*, do separacji poszczególnych wartości należy użyć spacji. Plik ma się składać z dwóch sekcji:

1. Informacje na temat walut, każda waluta w oddzielnym wierszu opisana trzema wartościami
 - indeks waluty - kolejne liczby naturalne, zaczynając od 0
 - 3-literowy kod waluty
 - pełna nazwa waluty
2. Informacje na temat kursów walut względem siebie, każdy kurs w oddzielnym wierszu opisana sześcioma wartościami:
 - indeks kursu - kolejne liczby naturalne, zaczynając od 0
 - kod waluty wejściowej
 - kod waluty wyjściowej
 - kurs
 - rodzaj prowizji - PROC - jeśli procentowa, STAŁA - jeśli stała
 - wysokość prowizji

Przykład:

0 BRL real brazylijski

1 PLN polski złoty

2 EUR euro

0 BRL PLN 0,9442 STAŁA 5

1 EUR PLN 4,2972 PROC 0,001

5 Obsługa błędów

Program będzie wyposażony w obsługę wyjątków spowodowanych błędnym podaniem argumentów wywołania i błędnych danych w pliku. Jeśli zajdzie sytuacja wyjątkowa zostanie wypisany odpowiedni komunikat. Przykładowe komunikaty o błędach to:

- "Podano nieprawidłową nazwę operacji. Jako pierwszy argument podaj *arbitraz* lub *wymiana*"
- "Nie znaleziono lub nie udało się otworzyć pliku"

- "Podana nazwa waluty nie istnieje"
- "Żądana waluta nie istnieje w zbiorze danych"
- "Plik zawiera nieprawidłowe dane w lini:"

6 Opis wymagań niefunkcjonalnych

Poza przejrzystością i łatwością obsługi jaką gwarantuje obsługa błędów program powinien działać odpowiednio krótko. Pod tym słowem rozumie się czas wykonania obliczeń rzędu kilku sekund dla niewielkich zbiorów danych tzn. kilkunastu walut. Dla dużych zbiorów danych, typu kilkaset walut z dużą liczbą połączeń czas wykonania się programu nie powinien przekraczać kilkunastu minut.