

Politechnika Warszawska
Wydział Elektryczny
Kierunek Informatyka

SPECYFIKACJA FUNKCJONALNA

Wykonał: Aliaksandr Karolik (295138)
Warszawa, 10.11.2018

Contents

1	Teoretyczny wstęp	2
2	Wymagania funkcjonalne	2
3	Wymagania niefunkcjonalne	3
4	Jak korzystać z programu	3
5	Komunikaty o błędach	5
6	Testy akceptacyjne	5

1 Teoretyczny wstęp

Ogólny pomysł tego programu jest taki "Jak zdobyć maksymalny zysk z powodu różnicy kursów wymiany walut przy jakimś kapitale startowym". Wymienić walutę można na dwa sposoby:

1. **Bezpośrednia wymiana waluty**
2. **Pośrednia wymiana waluty**

Przykładem wymiany bezpośredniej jest wymiana euro na dolary amerykańskie według kursu. Przykładem pośredniej wymiany waluty jest ciąg wymian. Na przykład euro na funty brytyjskie a później na dolary amerykańskie.

Co to jest arbitraż ekonomiczny? Arbitrażem ekonomicznym nazywamy zakup dobra lub aktywa na rynku, na którym jego cena jest względnie niższa w celu jednoczesnej odsprzedaży tego dobra lub aktywa na innym rynku, na którym jego cena jest względnie wyższa (źródło Wikipedia). W przypadku programu jest to wyszukiwanie najkorzystniejszej ścieżki wymiany walut, aby ostatecznie otrzymać większą kwotę w walucie wejściowej niż kwota którą miał użytkownik na początku. Przykład:

1000 EUR->GBP->USD->EUR->1000,25

2 Wymagania funkcjonalne

1. Znaleźć najkorzystną ścieżkę wymiany waluty,
2. Znaleźć dowolny arbitraż.

Co jest potrzebne aby program działał w trybie znalezienia najkorzystniejszej ścieżki wymiany waluty:

1. Podanie przez użytkownika pliku z walutami oraz kursami wymiany tych walut,
2. Podanie przez użytkownika argumentu decydującego o trybie pracy programu,
3. Podanie przez użytkownika kwoty pieniężnej,
4. Podanie przez użytkownika waluty wejściowej,

-
5. Podanie przez użytkownika waluty wyjściowej.

Co jest potrzebne aby program działał w trybie znalezienia dowolnego arbitrażu:

1. Podanie przez użytkownika pliku z walutami oraz kursami wymiany tych walut,
2. Podanie przez użytkownika argumentu decydującego o trybie działania programu,
3. Podanie przez użytkownika kwoty pieniężnej.

3 Wymagania niefunkcjonalne

Aby móc uruchomić program użytkownik musi zainstalować zintegrowane środowisko programistyczne IntelliJ IDEA dla Javy. Dla poprawnego działania programu są następujące minimalne wymagania systemowe:

1. Windows 10/8/7/Vista/XP (incl. 64-bit),
2. 2 GB RAM minimum, 4 GB RAM proponowane,
3. 1.5 GB wolnego miejsca na dysku,
4. co najmniej 1 GB pamięci podręcznej.

To są minimalne wymagania systemowe dla zintegrowanego środowiska programistycznego IntelliJ IDEA. Im więcej użytkownik chce przetwarzać informacji tym więcej zasobów będzie zurzywać program.

4 Jak korzystać z programu

Program będzie napisany w IntelliJ IDEA zintegrowanym środowisku programistycznym (IDE) dla Javy. Aby uruchomić program trzeba otworzyć projekt w IntelliJ IDEA a później korzystając z przyciska RUN uruchomić program. Program nie jest interaktywny. Wszystkie ustawienia programu podawane są jako argumenty.

Lista argumentów:

1. Nazwa pliku z danymi opisującymi waluty oraz kursy ich wymiany.
Możliwy format: *.txt.
2. Argument decydujący o trybie pracy programu. Dostępne tryby to:
 - (a) Tryb wyszukiwania najkorzystniejszej wymiany waluty. Aby uruchomić ten tryb należy podać argument **"bw"**,
 - (b) Tryb wyszukania dowolnego arbitrażu. Aby uruchomić ten tryb należy podać argument **"ar"**.

W przypadku gdy zostanie wybrany tryb wyszukiwania najkorzystniejszej wymiany waluty użytkownik musi podać następujące argumenty:

- (a) Kwota pieniężna którą użytkownik chce wymienić,
- (b) Trzy literowy skrót waluty którą użytkownik chce wymienić.
Przykład: **"USD"**,
- (c) Trzy literowy skrót waluty do której użytkownik chce wymienić podaną kwotę.
Przykład: **"EUR"**.

W przypadku gdy zostanie wybrany tryb wyszukiwania dowolnego arbitrażu użytkownik musi podać następujący argument:

- (a) Kwota pieniężna dla której użytkownik chce znaleźć arbitraż.

Przykładowe podanie argumentów dla trybów działania programu:

1. Użycie trybu wyszukiwania najkorzystniejszej wymiany waluty:

Przykładowy_plik.txt bw 1000 USD EUR

W wyniku wywołania programu z tymi argumentami, waluty oraz kursy walut zostaną wczytane z pliku "Przykładowy_plik.txt". W wyniku działania programu zostanie wyświetlony w konsoli najkorzystniejszy ciąg wymian dla waluty "USD" wymienionej do "EUR" dla kwoty 1000.

2. Użycie trybu wyszukiwania dowolnego arbitrażu:

Przykładowy_plik.txt ar 1000

W wyniku wywołania programu z tymi argumentami, waluty oraz kursy

walut zostaną wczytane z pliku "Przykładowy_plik.txt". W wyniku działania programu zostanie wyświetlony w konsoli dowolny arbitraż dla kwoty 1000.

5 Komunikaty o błędach

Program będzie obsługiwał następujące sytuacje wyjątkowe:

- Nie wystarczająca liczba argumentów - program wyświetli komunikat o niepoprawnej liczbie argumentów. Przykładowy komunikat: "Została podana nie poprawna ilość argumentów."
- Nie poprawny argument podany przez użytkownika - program wyświetli komunikat z informacją o błędnym argumencie. Przykładowy komunikat: "Został podany błędny argument:[błędny argument]. Proszę o sprawdzenie podawanych argumentów."
- Plik wejściowy nie istnieje, bądź brakuje dostatecznych uprawnień do otwarcia pliku - program wyświetli komunikat o braku pliku lub niedostatecznych uprawnieniach do otwarcia pliku. Przykładowy komunikat: "Nie udało się otworzyć plik o nazwie:[nazwa pliku]."
- Plik wejściowy zawiera niepoprawne dane - program wyświetli komunikat o niepoprawnych danych w pliku wejściowym. Przykładowy komunikat: "Plik o nazwie: [nazwa pliku] zawiera błąd w linii:[numer linii]."

6 Testy akceptacyjne

1. Test poprawności wyszukiwania najkorzystniejszej ścieżki dla wymiany waluty.

(a) Test dla poprawnych danych.

Test będzie przeprowadzony na danych w których już wiadomo jaka ścieżka wymian waluty jest najlepsza. Test zostanie zaakceptowany w przypadku gdy program wyświetli w konsoli tą ścieżkę. W innych przypadkach będzie stwierdzono że program nie spełnia wymagań funkcjonalnych.

(b) Test dla danych niepoprawnych.

Test zostanie przeprowadzony na danych które światomie nie będą miały nakożystniejszej wymiany waluty. Test zostanie zaakceptowany w przypadku gdy program wyświetli w konsoli komunikat o tym, że dla wejściowych danych nie udało się znaleźć najkożystniejszej wymiany waluty. W innych przypadkach zostanie stwierdzono że program nie spełnia wymagań funkcjonalnych.

2. Test poprawności wyszukiwania dowolnego arbitrażu.

(a) Test dla poprawnych danych.

Test zostanie przeprowadzony na danych w których jest możliwość znalezienia tylko jednego arbitrażu. Test zostanie zaakceptowany gdy program wyświetli ten arbitraż w konsoli. Jeżeli program zwróci nie poprawną ścieżkę albo zwróci komunikat że szukanie arbitrażu skończyło się niepowodzeniem, zostanie stwierdzono że program nie spełnia wymagań funkcjonalnych.

(b) Test dla nie poprawnych danych.

Test zostanie przeprowadzony dla danych które nie będą mieli żadnego abitrażu dla pliku wejściowego. Test zostanie zaakceptowany w przypadku gdy program wyświetli w konsoli komunikat o tym, że dane wejściowe nie mają żadnego arbitrażu. W innym przypadku będzie stwierdzone że program nie spełnia swoich wymagań funkcjonalnych.

3. Test odporności programu na błędy użytkownika.

Dla sprawdzenia czy program może weryfikować błędne dane które może podać użytkownik. Będzie przeprowadzono kilka scenariuszów działania programu:

(a) Podanie pliku który zawiera błędne formatowanie,

(b) Podanie przez użytkownika ujemnej kwoty pieniężnej albo nie podanie jej w cale,

(c) Podanie waluty której nie ma w pliku wejściowym,

(d) Podanie nie poprawnych argumentów decydujących o trybie działania programu.

Testy zostaną zaakceptowane gdy program zweryfikuje wszystkie błędy i wyświetli odpowiednie komunikaty do konsoli. W odwrotnym przypadku będzie stwierdzono że program nie spełnia wymagań funkcjonalnych.