

Zadanie 5

Jaś Roztargniony i duplikaty tekstowe



Opis

Jeden z naszych pracowników - Jaś Roztargniony - został poproszony o stworzenie modułu do nowo tworzonego systemu wyszukiwania duplikatów dokumentów tekstowych zamówionego przez *Instytut Magii Wysokich Energii*. Celem aplikacji jest wychwytywanie prac nieuczciwych studentów. Zadanie Jasia polegało na stworzeniu modułu mierzącego podobieństwo dwóch słów, wyrażonych łańcuchami znaków. Do



zmierzenia podobieństwa Jaś wykorzystał dwie znane miary odległości – odległość *Levenshteina*¹ oraz *Jaro-Winklera*². Kod dostarczony przez Jasia wydaje się działać poprawnie oraz wydajnie. Niestety jest nieczytelny. Wstępna recenzja dokonana przez Klienta potwierdziła obawy. Klient stwierdził, że jego zespół utrzymania nie będzie wpierał takiej aplikacji. Możemy sobie nawet pozwolić na pewne straty wydajności, ale oprogramowanie musi być czytelne **na pierwszy rzut oka** dla inżyniera utrzymania. Termin oddania projektu już za tydzień a Jaś, niestety wdraża konwerter liczb w NASA przez co jest nieosiągalny. Jedyna nadzieja, że to Ty nam pomożesz!

Opis zadania

Uczestnicy mają na celu refactoring dostarczonego kodu. Może mu podlegać wyłącznie implementacja.

API zostawiamy bez zmian.

Refaktoryzując kod kieruj się następującymi zasadami:

- SOLID
- Tell, don't ask
- Prawo Demeter

oraz innymi dobrymi praktykami, które znasz.

¹ http://en.wikipedia.org/wiki/Levenshtein_distance

² http://en.wikipedia.org/wiki/Jaro-Winkler_distance



Wejście

Kod w jednym z języków Java/C++/C# .

- C++

mastercoder.pl/download/zad5/c/text-similarity-calculator.zip

- C#

mastercoder.pl/download/zad5/csharp/text-similarity-calculator.zip

- Java

mastercoder.pl/download/zad5/java/text-similarity-calculator.zip

Rozwiązanie

W ramach refactoringu możesz **modyfikować istniejący kod, tworzyć nowe klasy oraz metody jak również usuwać już istniejące**. Na pewno docenimy fakt, iż upewnisz się, że kod działa tak samo przed jak i po refactoringu. Możesz w tym celu zadbać o pokrycie kodu testami jednostkowymi. Część z nich Jaś po sobie zostawił. Jeśli uważasz, że ich brakuje, zachęcamy do ich uzupełnienia.

Jaś zapewniał, że jego implementacja jest bezbłędna, jeśli jednak znajdziesz błąd programie lub testach, zostaną Ci przyznane dodatkowe punkty.

Jesteśmy również bardzo ciekawi motywów, jakie kierują Tobą, gdy podejmujesz swoje decyzje dotyczące refactoringu, dlatego docenimy również krótki opis typu:

Zmiana

Zastąpienie instrukcji „if... else...” w linii 45 polimorfizmem

Uzasadnienie

Zamknięcie implementacji z bloków tych instrukcji we wzorcu projektowym strategii. Zapewniamy spełnienie Open-Closed principle. W tej chwili możemy tworzyć nowe strategie, które będą tworzone przez fabrykę bez konieczności modyfikacji kodu biznesowego.

Ważne terminy

- Publikacja zadania: 25-03-2014
- Ostateczny termin nadsyłania odpowiedzi: 30-03-2014 godz.: 23:59
- Ogłoszenie wyników: 07-04-2014

Ocenianie

Zadanie będzie oceniane według poniższych kryteriów:

- Oceniamy osobno każdą decyzję o refactoringu i jej uzasadnienie. Za każdą zmianę możemy przyznać **do 0 do 60 punktów** (w zależności od rodzaju wprowadzonej zmiany). Maksymalna ilość punktów za refactoring: **210 pkt.**
- Za zaskakujące i nietypowe rozwiązanie, za znalezienie błędów tudzież wszystko co nas pozytywnie zaskoczy: **do 90 punktów.**

Maksymalna ilość punktów do uzyskania w zadaniu: **300 pkt.**³

Złote zasady i ważne informacje

- przed wysłaniem sprawdź, czy kod się kompiluje
- nie zmieniaj API
- źródła przesyłaj jako załącznik do e-maila skompresowany za pomocą ZIPa, 7ZIP bądź RAR - możliwe też jest umieszczenie rozwiązania jako publicznego zasobu np.: na OneDrive, GoogleDrive. **Jeśli nie uzyskasz od nas potwierdzenia wysłania swojego zadania w przeciągu 24 godzin daj nam znać. Późniejsze reklamacje nie będą uwzględniane.**
- swoje rozwiązanie prześlij na: **mastercoder.poland@cybercom.com**
- w temacie e-maila z odpowiedzią podaj język, technologię za pomocą której problem został rozwiązany, przykładowo: [Re: MasterCoder Zadanie 5 Duplikaty tekstowe - rozwiązanie C#]
- w razie pytań pisz na wyżej wymieniony adres e-mail
- dobrze się baw!!!

³ Rozwiązania dla których wszystkie dostarczone testy nie zakończą się po 600 sekundach będą otrzymywały 0 punktów. Maszyna referencyjna: Core: i5 2,5GHz, RAM: 4GB

