

Implementacja algorytmu Insertion Sort w podejściu TDD

Grzegorz Rozdzialik

22 maja 2021

Streszczenie

Sprawozdanie zadania 1. zawiera pierwsze 3 cykle podejścia TDD (ang. Test-Driven Development) dla implementacji algorytmu Insertion Sort w języku Rust.

Jest to rozwiązanie zadania 1. z przedmiotu Testowanie i Weryfikacja Oprogramowania studiach magisterskich OKNO 2020/2021.

Spis treści

Opis algorytmu Insertion Sort	1
Podejście Test-Driven Development	1
Implementacja algorytmu z zadania	2
Cykl 1	2
Cykl 2	2
Cykl 3	2
Bibliografia	2

Opis algorytmu Insertion Sort

Algorytm Insertion Sort ma na celu posortowanie listy elementów. Realizuje to poprzez podzielenie listy na 2 podlisty:

1. Lista elementów już posortowanych
2. Lista elementów oczekujących na posortowanie

A następnie wybieraniu kolejnych elementów z listy elementów oczekujących na posortowanie i umieszczaniu ich w odpowiednim miejscu w liście elementów posortowanych.

Początkowo lista elementów posortowanych zawiera wyłącznie pierwszy element listy, a pozostałe są uznawane jako nieposortowane.

Szerszy opis oraz przykład działania algorytmu został zaprezentowany na stronie internetowej https://www.tutorialspoint.com/data_structures_algorithms/insertion_sort_algorithm.htm

Podejście Test-Driven Development

W podejściu Test-Driven Development (TDD) implementacja algorytmu lub funkcjonalności następuje w kolejno następujących po sobie cyklach składających się z następujących kroków:

1. Napisanie testu dla niezrealizowanej funkcjonalności. Wykonanie tego testu powinno zakończyć się porażką.
2. Implementacja funkcjonalności. Po zakończeniu implementacji nowo napisany test, jak i testy napisane w poprzednich cyklach powinny zakończyć się powodzeniem.

3. Czyszczenie, ulepszenie implementacji (ang. refactoring). Po tym kroku wszystkie testy nadal powinny kończyć się powodzeniem.

Implementacja algorytmu z zadania

Algorytm z zadania zaimplementowano w języku Rust (<https://www.rust-lang.org/>).

Do uruchomienia testów użyto standardowej komendy dla języka Rust:

```
cargo test
```

W kolejnych sekcjach zostaną zaprezentowane kroki wykonane dla pierwszych 3 cykli implementacji algorytmu w podejściu TDD.

Cykl 1

Cykl 2

Cykl 3

Bibliografia

1. Opis algorytmu Insertion Sort
https://www.tutorialspoint.com/data_structures_algorithms/insertion_sort_algorithm.htm
2. Slajdy do przedmiotu Testowanie i Weryfikacja Oprogramowania na studiach magisterskich OKNO
3. Dokumentacja pisania testów w języku Rust <https://doc.rust-lang.org/book/ch11-01-writing-tests.html>