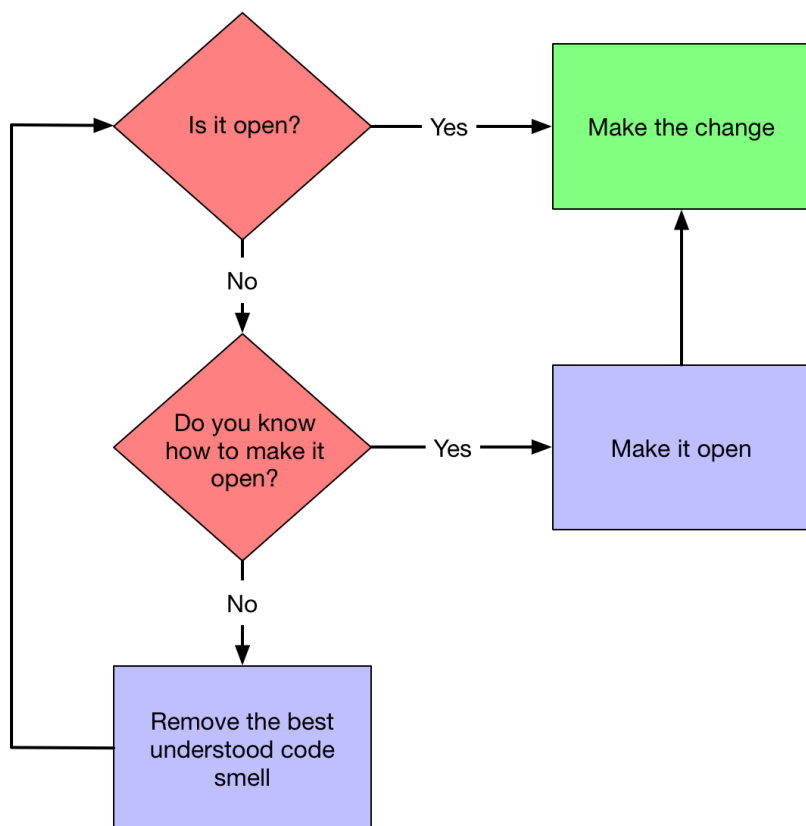


# Open/Closed Principle

Włodek Bzyl

Version 1.1, December 11, 2017

Na egzamin należy zaproponować zmianę w istniejącym kodzie. Następnie poprzez kolejne refaktoryzacje, zmieniamy kod, tak aby otworzyć go na proponowaną zmianę. Aby uzyskać zamierzoną otwartość w kodzie można posłużyć się poniższym schematem blokowym.



Source: *99 Bottles of OOP* by S. Metz & K. Owen

W trakcie refaktoryzacji obowiązują następujące reguły:



1. Zmiany w kodzie dotyczą tylko jednego wiersza (jednej instrukcji).
2. Uruchamiamy testy po każdej zmianie.
3. Jeśli jakiś test nie zostanie zaliczony, to wracamy do poprzedniej wersji i wykonujemy inną lepszą zmianę.

Przykłady kodu z propozycjami zmian – [RinR](#), [:ant](#): (brak testów i ewentualnych zmian), [Conway's Game of Life](#) (zmiana: użyć biblioteki [Curses](#); zob. też [mmozuras](#)).

Kilka odsyłaczy ilustrujących lub opisujących o co chodzi w zasadzie Open/Closed:

- Jordan Hudgens, [SOLID Development: Open Closed Principle Guide and Example](#)
- Robert C. Martin, [The Open Closed Principle](#)

- Sergii Makagon, [S\[O\]LID - Open/Closed Principle by example](#)

# Organizacja kodu i dokumentacji w repozytoriach

W repozytoriach należy umieścić:

1. plik *README* (w notacji Markdown lub ADOC) w którym:
  - a. opisano (krótko) kod oraz proponowaną zmianę
  - b. w opisano zapachy w kodzie i refaktoryzacje, które doprowadziły do zamierzonej otwartości w kodzie
  - c. opisano proces za pomocą którego wprowadzono proponowaną zmianę
2. plik *Gemfile* z listą użytych gemów
3. w podkatalogu *lib* kod w Ruby oraz plik *.reek.todo* ignorujący zapachy w kodzie nieopisane w książce M. Fowler „Refaktoryzacja”, na przykład `IrresponsibleModule`, `DuplicateMethodCall`, `UncommunicativeVariableName`
4. w podkatalogu *spec* testy RSpec; albo w katalogu *test* testy MiniSpec (lub inne, np. TestUnit)

Plik *README* należy skonwertować do formatu PDF, dodać do repozytorium, wydrukować i dostarczyć przed upływem wyznaczonego terminu.

Jeśli plik jest w formacie [AsciiDoctor](#), to do konwersji można użyć programu *asciidoctor-pdf* (zainstalowanego na serwerze Sigma):

```
asciidoctor-pdf README.adoc
```

Zob. [egzamin\\_template.pdf](#) – przykładowy dokument po konwersji z formatu ADOC do PDF.