Software Engineering

WS 2022/23, Sheet 03

Prof. Dr. Sven Apel
Yannick Lehmen, Maurice Vincon

Handout: 14.11.2022

Task 1

Discussion: Prepare arguments for and against implementing variability by using runtime parameters. You can present your arguments during the discussion in the tutorial sessions.

Solution

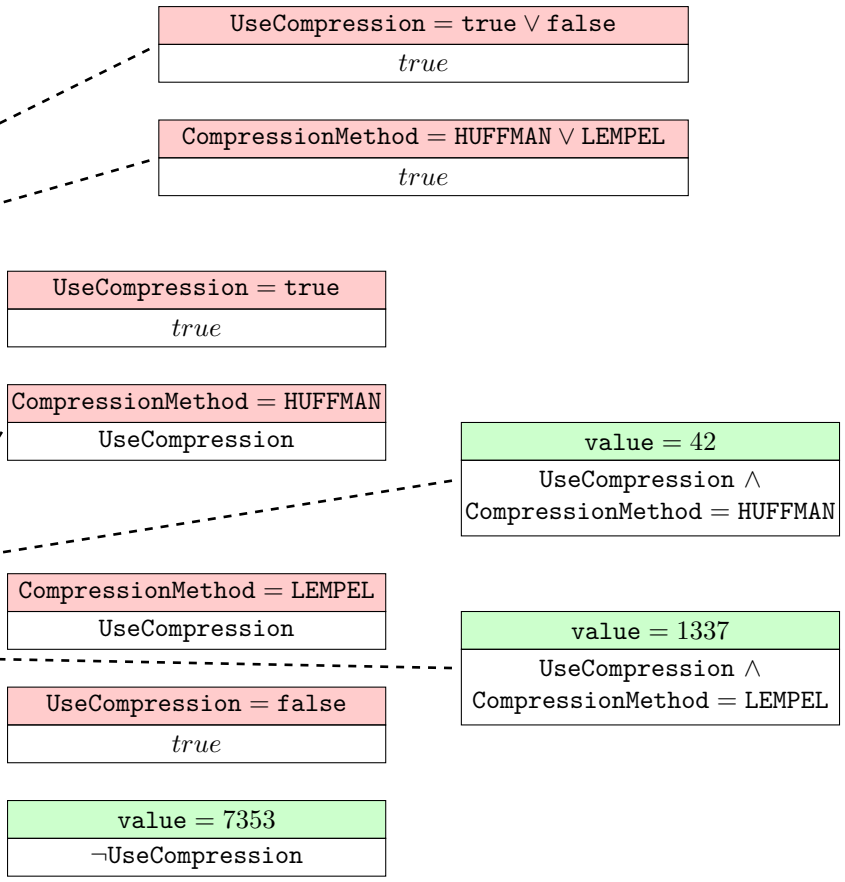
Task 2

Assuming that the struct `Config` contains feature variables, perform a taint analysis on the function `foo()` in the following code example. It is sufficient to do this intuitively as it was done in the example in the lecture; you do not need to follow the formal algorithm.

```
enum class CompressionMethod {
  HUFFMAN, LEMPEL
}

struct Config {
  static bool UseCompression;
  static CompressionMethod CompressionMethod;
};

int foo() {
  int value;

  if (Config.UseCompression) {
    switch (Config.CompressionMethod) {
    case CompressionMethod::HUFFMAN:
      value = 42; break;
    case CompressionMethod::LEMPEL:
      value = 1337; break;
    }
  } else {
    value = 7353;
  }
  return value;
}
```

Solution

```
1  enum class CompressionMethod {
2    HUFFMAN, LEMPEL
3  }
4
5  struct Config {
6    static bool UseCompression;
7    static CompressionMethod
       ↪  CompressionMethod;
8  };
9
10 int foo() {
11   int value;
12
13   if (Config.UseCompression) {
14     switch (Config.CompressionMethod)
         ↪  {
15     case CompressionMethod::HUFFMAN:
16       value = 42; break;
17     case CompressionMethod::LEMPEL:
18       value = 1337; break;
19     }
20   } else {
21     value = 7353;
22   }
23   return value;
24 }
```

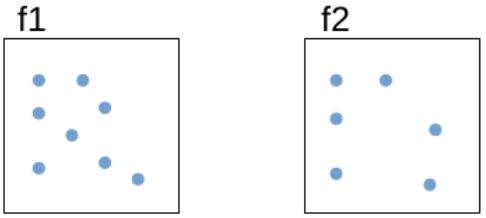


Task 3

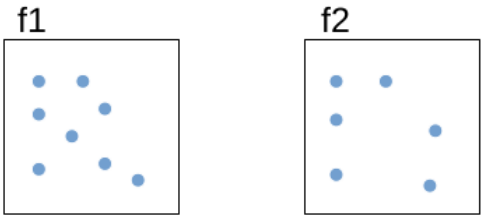
a) Define the terms *Cohesion* and *Coupling*.

b) Complete the diagrams such that they represent the following combinations of cohesion and coupling between the features $f1$ and $f2$:

(i) High cohesion, strong coupling.



(ii) Low cohesion, weak coupling.

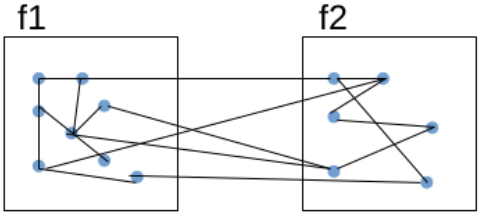


c) Calculate the Internal Feature Dependency IFD and the External Feature Dependency EFD for $f1$ and $f2$ for both cases from b).

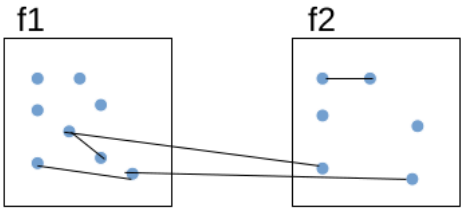
Solution

a) *Cohesion* is a metric of how connected code is within a module. High cohesion is usually desirable. *Coupling* is a metric of how connected different modules are among each other. The goal is to have low coupling.

b) (i) High cohesion, strong coupling.



(ii) Low cohesion, weak coupling.



c) (i)

$$\text{IFD}(f1) = \frac{8}{(8 \times 7)/2} = \frac{2}{7} \approx 0.29 \qquad \text{IFD}(f2) = \frac{5}{(6 \times 5)/2} = \frac{1}{3} \approx 0.33$$

$$\text{EFD}(f1) = \frac{5}{13} \approx 0.38 \qquad \text{EFD}(f2) = \frac{5}{10} = 0.5$$

(ii)

$$\text{IFD}(f1) = \frac{2}{(8 \times 7)/2} = \frac{1}{14} \approx 0.07 \qquad \text{IFD}(f2) = \frac{1}{(6 \times 5)/2} = \frac{1}{15} \approx 0.07$$

$$\text{EFD}(f1) = \frac{2}{4} = 0.5 \qquad \text{EFD}(f2) = \frac{2}{3} \approx 0.66$$