Week 9: Exercises on Properties of Code

Multiple Choice Questions

Question 9M.1

Jessica writes the following program.

```
nat x = 0;
nat y = 2;
while (x < y) {
  x = x+1;
  y = y+2;
}
```

Bill writes the following program.

```
nat x = 0;
nat y = 2;
while (x < y) {
  x = x+2;
  y = y+1;
}
```

(a) Both programs halt.

(b) Both programs hang.

(c) Jessica's program halts and Bill's hangs.

(d) Jessica's program hangs and Bill's halts.

Question 9M.2

Let P be a decision problem. To prove P undecidable, it suffices to

(a) Reduce P to a decidable problem.

(b) Reduce P to an undecidable problem.

(c) Reduce a decidable problem to P.

(d) Reduce an undecidable problem to P.

Question 9M.3

Natalie, Harry and Xerxes each write a program with parameter `nat x`. Natalie's program is as follows.

```
nat y = x+2;
while (y < 100) {
  println("Hello");
  y = y+1;
}
```

Harry's program is as follows.

```
nat y = x;
while (y < 98) {
   println("Hello");
   println("Hello");
   y = y+2;
}
```

Xerxes' program is as follows.

```
nat y = x;
while (y < 98) {
   println("Hello");
   y = y+1;
}
```

(a) Natalie and Harry's program have the same semantics, but Xerxes' has a different semantics.

(b) Natalie and Xerxes' programs have the same semantics, but Harry's has a different semantics.

(c) Harry and Xerxes' programs have the same semantics, but Natalie's has a different semantics.

(d) All three programs have the same semantics.

Question 9M.4

The body-code of a method

```
nat f (nat x){
  ...
}
```

is *sleepy* when `f(5)` and `f(6)` and `f(7)` all return even numbers.

(a) Sleepiness is semantic and decidable.

(b) Sleepiness is semantic and undecidable.

(c) Sleepiness is not semantic and is decidable.

(d) Sleepiness is neither semantic nor decidable

Question 9M.5

Which of the following is true?

(a) Every undecidable property of code holds in some case.

(b) Every decidable property of code fails to hold in some case.

(c) Every semantic property of code holds in some case.

Exercises

Exercise 9.1

Consider the following properties of code.

1. A procedure

   ```
   void f (nat n)
   ```

   is *happy* when "a" has no more occurrences in the body code than "b".

2. A procedure

   ```
   void f (nat n)
   ```

   is *joyous* when, if you apply it to any even number, it prints at least 3 characters.

3. A procedure

   ```
   void f (nat n)
   ```

   is *glad* when either "a" has no more occurrences in the body code than "b", or "b" than "c", or "c" than "a".

4. A pair of methods

   ```
   nat f (nat n)
   nat g (nat n)
   ```

   is *jolly* when, for any integer n, either $\texttt{f}(n)$ hangs or $\texttt{g}(n)$ hangs, or they both terminate and the sum of the values returned is 33.

For each of these properties, answer the following questions, with explanation.

- Does it hold in some case?
- Does it fail to hold in some case?
- Is it semantic?
- Is it decidable?

Exercise 9.2

Dave manages a software firm. His employees write programs that interact with a user by inputting and outputting text. Dave tells them that, as long as the user has only input basic English words, their code should only output basic English words. (Here, "basic" means no longer than 8 characters.) Dave wants a tool to check that his employees' programs meet this requirement. Is that possible? Explain your answer.

Exercise 9.3

Sketch a procedure which tests whether a given polynomial (with integer coefficients the first coefficient being 1) has an integer root. So for example, for $x^3 - 4x^2 + x + 6$ the answer should be "yes", because 3 is such a root, and for $x^2 - 2$ the answer should be no, because the only two roots, $\sqrt{2}$ and $-\sqrt{2}$, are irrational.