

Exercise Sheet 9 - Model Answers

1. (a) A number m greater than 1 and smaller than n that divides n .
- (b) An assignment of a truth value to each propositional variable occurring in ϕ such that ϕ evaluates to true.
- (c) An enumeration of the nodes of the tree that corresponds to a breadth-first traversal. The checker only has to test whether indeed every node (except the first) has exactly one other node earlier in this enumeration to which it is connected.

2. (a)

$$p_{1a} \vee p_{1b} \vee p_{1c} \vee p_{1d} \vee p_{1e}$$

(b)

$$\begin{aligned} & \neg(p_{1a} \wedge p_{1b}) \wedge \neg(p_{1a} \wedge p_{1c}) \wedge \neg(p_{1a} \wedge p_{1d}) \wedge \neg(p_{1a} \wedge p_{1e}) \\ & \wedge \neg(p_{1b} \wedge p_{1c}) \wedge \neg(p_{1b} \wedge p_{1d}) \wedge \neg(p_{1b} \wedge p_{1e}) \\ & \wedge \neg(p_{1c} \wedge p_{1d}) \wedge \neg(p_{1c} \wedge p_{1e}) \\ & \wedge \neg(p_{1d} \wedge p_{1e}) \end{aligned}$$

(c)

$$p_{1a} \vee p_{2a} \vee p_{3a} \vee p_{4a} \vee p_{5a}$$

(d)

$$\begin{aligned} & (p_{2a} \rightarrow p_{3b}) \\ & \wedge (p_{2b} \rightarrow p_{3a} \vee p_{3c} \vee p_{3d} \vee p_{3e}) \\ & \wedge (p_{2c} \rightarrow p_{3b} \vee p_{3d}) \\ & \wedge (p_{2d} \rightarrow p_{3b} \vee p_{3c}) \\ & \wedge (p_{2e} \rightarrow p_{3b}) \end{aligned}$$

3. The factorization problem is not a decision problem, because a decision problem has to have answer Yes or No.

Checking that y is a witness is easy: do the division x/y to check that it is a factor, and calculate the rightmost bits to check they agree with z . Both these parts are linear.

The case $m = z = 0$ is simply to check whether x has a factor, and we have the polynomial AKS algorithm.

Here is the reduction. We can write it as pseudocode. $F(x, m, z)$ is the decision problem described in the question. n is the number of bits of x .

```

if (!F(x, 0, 0)) return "No factor";

int m = 0;
int z = 0;

/* invariant:
 * 0 <= m <= n and
 * x has a factor y that agrees with z on rightmost m bits
 */
while (m < n) {
    if (!F(x, m+1, z)) {

```

```

    z = z + 2^m;
  }
  m = m+1;
}
return z;

```

This has polynomial degree (in n) one more than that of the decision problem.

4. The number of steps is $2^n - 1$. To prove this, from the equation we get

$$f(n+1) + 1 = 2f(n) + 2 = 2(f(n) + 1)$$

and so $f(n) + 1$ is doubled each time n increases by 1. Since $f(1) + 1 = 2 = 2^1$, we see that $f(n) + 1 = 2^n$.

The number of steps to process an input of size 64 is

$$2^{64} = 2^4 \times (2^{10})^6 \approx 16 \times 10^{18}$$

Hence the time taken is approximately

$$\begin{aligned}
 \frac{16 \times 10^{18}}{10^9} &= 16 \times 10^9 \text{ seconds} \\
 &= \frac{16 \times 10^9}{30 \times 10^6} \text{ years} \\
 &\approx 500 \text{ years.}
 \end{aligned}$$

(One story said that monks in Hanoi were moving the 64 discs at a rate of one per second and that when they finished the world would end. They will take a billion times as long as our computer.)