Solutions of Assignment 12

Theory of Computation, Fall 2022

Keys

Q1

$$div(m,0) = m, \quad m \in \mathcal{N} \tag{1}$$

Note: The proof is as follows.

The definition of $rem(m,n),\ m,n \in \mathcal{N}$ is as follows:

$$\begin{aligned} rem(0,n) &= 0 \\ rem(m+1,n) &= \begin{cases} 0, & \text{if } eq(rem(m,n), pred(n)) \\ rem(m,n)+1, & \text{otherwise} \end{cases} \end{aligned} \tag{2}$$

We learn that:

- $rem(0,0) = 0$.
- Assume that $rem(k,0) = 0$, there is $rem(k+1,0) = 0$ since $pred(0) = 0$ and $eq(rem(k,0), pred(0)) = 1$.

We can conclude that $rem(m,0) = 0,\ m \in \mathcal{N}$.

The definition of $div(m,n),\ m,n \in \mathcal{N}$ is as follows:

$$\begin{aligned} div(0,n) &= 0 \\ div(m+1,n) &= \begin{cases} div(m,n)+1, & \text{if } eq(rem(m,n), pred(n)) \\ div(m,n), & \text{otherwise} \end{cases} \end{aligned} \tag{3}$$

We learn that:

- $div(0,0) = 0$.
- $\forall m \in \mathcal{N},\ eq(rem(m,0), pred(0)) = 1 \implies div(m+1,0) = div(m,0)+1$.

We can conclude that $div(m,0) = m,\ m \in \mathcal{N}$.

Q2

Proof:

Define $g : \mathcal{N} \times \mathcal{N} \to \mathcal{N}$ to be

$$g(m, n) = f(f(\ldots f(n)\ldots)) \tag{4}$$

where there are m compositions.

g can be written as

$$\begin{aligned} g(0, n) &= f(n) \\ g(m+1, n) &= f(g(m, n)) \end{aligned} \tag{5}$$

Since f is primitive recursive, so is g.

We have that

$$F(n) = g(n, n) \tag{6}$$

that is,

$$F(n) = g(id_{1,1}(n), id_{1,1}(n)) \tag{7}$$

Therefore, F is the composition of primitive recursive functions. Thus F is primitive recursive.

> **Note:** The proof based on that "F is the composition of n primitive recursive functions f, so F is primitive recursive" is incorrect, because n is not a constant. You may refer to the proof showing that $sum_f(m, n)$ is primitive recursive on class (W12) to see why.

Q3

Proof:

Fix an arbitrary $k \geq 2$. For $i \in [1, k]$, define P_i as follows.

$$P_i(n_1, \dots, n_k) = \begin{cases} 1, & \text{if } (n_i = \max\{n_1, \dots, n_k\}) \wedge (\forall j < i, n_j \neq \max\{n_1, \dots, n_k\}) \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

P_i is a primitive recursive predicate since P_i can be written as

$$P_i(n_1, \dots, n_k) = (n_i > n_1) \wedge \dots \wedge (n_i > n_{i-1}) \wedge (n_i \geq n_{i+1}) \wedge \dots \wedge (n_i \geq n_k) \tag{9}$$

Now φ_k can be written as follows.

$$\varphi_k(n_1, \dots, n_k) = \sum_{i=1}^{k} P_i(n_1, \dots, n_k) \cdot n_i \tag{10}$$

Therefore, φ_k is the composition of primitive recursive functions. Thus φ_k is primitive recursive.

Another Proof:

φ_2 can be written as follows.

$$\varphi_2(n_1, n_2) = \max\{n_1, n_2\} = \begin{cases} id_{2,1}(n_1, n_2), & \text{if } \ geq(n_1, n_2) \\ id_{2,2}(n_1, n_2), & \text{otherwise} \end{cases} \tag{11}$$

Since *geq* is primitive recursive, so is φ_2.

Fix an arbitrary $k \geq 2$. φ_k can be written as follows.

$$\varphi_k(n_1, \dots, n_k) = \varphi_2(n_1, \varphi_2(n_2, \dots \varphi_2(n_{k-1}, n_k) \dots)) \tag{12}$$

Therefore, φ_k is the composition of primitive recursive functions. Thus φ_k is primitive recursive.

> **Note:** Here k is a fixed constant, so **another proof** is correct.

Q4

Proof:

h_p can be written as follows.

$$h_p(n) = \prod_{t=0}^{n} p(t) = mult_p(n) \tag{13}$$

> **Note:** Here $mult_p$ is defined as follows.
>
> $$\begin{aligned} mult_p(0) &= p(0) \\ mult_p(n+1) &= mult_p(n) \cdot p(succ(n)) \end{aligned} \tag{14}$$
>
> Therefore, $mult_p$ is primitive recursive.

Then h_p is the composition of primitive recursive functions. Thus h_p is primitive recursive.

Grading

100 pts. in total

- Q1. 10 pts.
- Q2. 40 pts.
- Q3. 30 pts.
- Q4. 20 pts.
