Solution to Problem 4

Problem. Write a Java program corresponding to the following primitive recursive function $f = \sigma(PR(\pi^2_2, \sigma(add(\pi^4_1, \pi^4_2, \pi^4_3))))$. For this function f, what is the value of $f(0, 2, 2)$?

Solution. In general, the expression $h = PR(f, g)$ corresponding to functions $f(n_1, \ldots, n_k)$ and $g(n_1, \ldots, n_k, m, h)$ defines a function of $k + 1$ variables:

$$h(n_1, \ldots, n_k, 0) = f(n_1, \ldots, n_k);$$

$$h(n_1, \ldots, n_k, m + 1) = g(n_1, \ldots, n_k, m, h(n_1, \ldots, n_k, m)).$$

In our cases, $f = \pi^2_2$ is a function of 2 variables, so $k = 2$. For $k = 2$, the general formulas for primitive recursion take the following form:

$$h(n_1, n_2, 0) = f(n_1, n_2);$$

$$h(n_1, n_2, m + 1) = g(n_1, n_2, m, h(n_1, n_2, m)).$$

Here, $f(n_1, n_2) = \pi^2_2(n_1, n_2) = n_2$ and

$$g(n_1, n_2, m, h) = \sigma(add(\pi^4_1(n_1, n_2, m, h), \pi^4_2(n_1, n_2, m, h), \pi^4_3(n_1, n_2, m, h))) =$$

$$\sigma(add(n_1, n_2, m)) = n_1 + n_2 + m + 1.$$

Thus, we have

$$h(n_1, n_2, 0) = n_2;$$

$$h(n_1, n_2, m + 1) = n_1 + n_2 + m + 1.$$

Primitive recursion is the description of a for-loop. The first line of the primitive recursion describes what is happening before the loop. In Java, the corresponding statement takes the following form:

```
int h = n2;
```

The second line of the primitive recursion describes what happens when we get from the iteration number $i - 1 = m$ to iteration number $i = m + 1$. So, we take

```
h = n1 + n2 + i;
```

The whole code for the PR part takes the form:

```
int h = n2;
for(int i = 1; i <= m; i++)
  {h = n1 + n2 + i;}
```

The desired function f is obtained from the PR expression by applying σ, i.e., by adding 1. Thus, we have the following Java program for computing the function f:

```
int h = n2;
for(int i = 1; i <= m; i++)
  {h = n1 + n2 + i;}
h++;
```

Let us trace this Java program on the example of $n1 = 0$, $n2 = 2$, and $m = 2$.

- We start with assigning, to the variable h, the value $n1 = 0$.
- Then, we go into the for-loop, and define the new variable i whose value is 1.
- Here, $i = 1 \le m = 2$, so we go inside the loop, and assign, to the variable h, the new value $h = n1 + n2 + i = 0 + 2 + 1 = 3$.
- After that, we increase i by 1, so i is now 2.
- Here still, $i = 2 \le m = 2$, so we go inside the loop, and assign, to the variable h, the new value $h = n1 + n2 + i = 0 + 2 + 2 = 4$.
- After that, we increase i by 1, so i is now 3.
- For $i = 3$ and $m = 2$, the condition $i \le m$ is no longer satisfied, so we get out of the loop.
- Finally, we increase the value h by 1, getting $h = 5$.

The value 5 is the desired value of the function $f(0, 2, 2)$.

Comment. Instead of tracing the Java program, we can trace the original formulas for primitive recursion, which for $n_1 = 0$ and $n_2 = 2$, take the form

$$h(0, 2, 0) = 2;$$

$$h(0, 2, m + 1) = 0 + 2 + m + 1.$$

For $m = 0$, the second formula leads to

$$h(0, 2, 1) = 0 + 2 + 0 + 1 = 3.$$

For $m = 1$, this formula leads to

$$h(0, 2, 2) = 0 + 2 + 0 + 2 = 4.$$

Thus, in this case, $h = PR(\ldots) = 4$.

To get the value of the desired function $f = \sigma(PR(\ldots))$, we need to add 1 to the PR expression $PR(\ldots) = 4$, so the final answer is 5.