Solution to Problem 4

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

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 the general case, f is a function of k variable, and g is a function of $k+2$ variables.

In our cases, $g = sum(\pi^2_2, \pi^2_1)$ is a function of 2 variables, so $k + 2 = 2$ and thus, $k = 0$. For $k = 0$, the general formulas for primitive recursion take the following form:

$$h(0) = f();$$

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

Here, $f() = \sigma(0) = 1$ and

$$g(m, h) = sum(\pi^2_2, \pi^2_1) = sum(h, m) = h + m.$$

Thus, we have

$$h(0) = 1;$$

$$h(m + 1) = h(m) + m.$$

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 = 1;
```

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 = h + i - 1;
```

The whole code for the PR part takes the form:

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

The desired function F is obtained from the PR expression by applying σ twice. Thus, we have the following Java program for computing the function F:

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

Let us trace this Java program on the example of $m = 2$.

- We start with assigning, to the variable h, the value 1.
- Then, we go into the for-loop, and define the new variable i whose value is 1.
- Here, $i = 1 \leq m = 2$, so we go inside the loop, and assign, to the variable h, the new value $h = 1 + 1 - 1 = 1$.
- After that, we increase i by 1, so i is now 2.
- Here, $i = 2 \leq m = 2$, so we go inside the loop, and assign, to the variable h, the new value $h = 1 + 2 - 1 = 2$.
- After that, we increase i by 1, so i is now 3.
- Here, $i = 3 > m = 2$, so we get out of the loop.
- Now, we increase the value h by 1, getting $h = 3$.
- Finally, we increase the value h by 1, getting $h = 4$.

The value 4 is the desired value of the function $F(2)$.

Comment. Instead of tracing the Java program, we can trace the original formulas for primitive recursion, which takes the form

$$h(0) = 1;$$

$$h(m + 1) = h(m) + m.$$

For $m = 0$, we get $h(0) = 1$. For $m = 0$, the second formula leads to

$$h(1) = h(0) + 0 = 1 + 0 = 1.$$

For $m = 1$, we get

$$h(2) = h(1) + 1 = 1 + 1 = 2.$$

Thus, in this case, $h = PR(\ldots) = 2$.

To get the value of the desired function $F = \sigma(\sigma(PR(\ldots)))$, we need to add 1 twice to the PR expression $PR(\ldots) = 2$, so the final answer is 4.