Solution to Problem 4

Problem. Write a Java program corresponding to the following primitive recursive function

\[ F = \sigma(PR(\sigma(0)), sum(\pi^2_2, \sigma(\pi^2_1)))) \]

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:

\[
\begin{align*}
    h(n_1, \ldots, n_k, 0) &= f(n_1, \ldots, n_k); \\
    h(n_1, \ldots, n_k, m + 1) &= g(n_1, \ldots, n_k, m, h(n_1, \ldots, n_k, m)).
\end{align*}
\]

In the general case, \( f \) is a function of \( k \) variable, and \( g \) is a function of \( k + 2 \) variables.

In our cases, \( g = \text{mult}(\pi^2_2, \sigma(\pi^2_1)) \) 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:

\[
\begin{align*}
    h(0) &= f() \\
    h(m + 1) &= g(m, h(m)).
\end{align*}
\]

Here, \( f() = \sigma(\sigma(0)) = 2 \) and

\[
    g(m, h) = \text{sum}(\pi^2_2, \sigma(\pi^2_1)) = \text{sum}(h, m + 1) = h + (m + 1).
\]

Thus, we have

\[
\begin{align*}
    h(0) &= 2; \\
    h(m + 1) &= h(m) + (m + 1).
\end{align*}
\]

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:

\[
\text{int h = 2;}
\]

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

\[
    h = h + i;
\]

The whole code for the \( PR \) part takes the form:
int h = 2;
for(int i = 1; i <= m; i++)
    {h = h + i;}

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

```java
int h = 2;
for(int i = 1; i <= m; i++)
    {h = h + i;}
h++;
```

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

- We start with assigning, to the variable $h$, the value 2.
- 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 $h = 2 + 1 = 3$.
- 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 $h = 3 + 2 = 5$.
- After that, we increase $i$ by 1, so $i$ is now 3.
- Here, $i = 3 > m = 2$, so we get out of the loop.
- Finally, we increase the value $h$ by 1, getting $h = 6$.

The value 6 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

$$h(0) = 2;$$

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

For $m = 0$, we get $h(0) = 2$. For $m = 1$, the second formula leads to

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

For $m = 1$, we get

$$h(2) = h(1) + (1 + 1) = 3 + 2 = 5.$$

Thus, in this case, $h = PR(...) = 5$.

To get the value of the desired function $F = \sigma(PR(...))$, we need to add 1 to the $PR$ expression $PR(...) = 5$, so the final answer is 6.