Question 1:

What function $\psi_p^{(2)}$ does the following program $P$ compute? Give a precise definition.

```
X1 ← X1 – 1
X1 ← X1 – 1
X1 ← X1 – 1
X2 ← X2 – 1
X2 ← X2 – 1
Y ← Y + 1
IF X1 ≠ 0 GOTO E
[A] Y ← Y + 1
IF X2 ≠ 0 GOTO A
```

Question 2:

a) Write a program $P$ in the language $L$ that computes the function $\psi_p^{(1)}(r) = 2r - 4$. You cannot use any macros at all, but only the three types of instructions in the original definition of the language. Note that we only use nonnegative integers (natural numbers), i.e., if $2r - 4$ is negative for a given input, the output is undefined and the program can never terminate.

b) Write down the complete list of snapshots that occur during the computation of $\psi_p^{(1)}(3)$. Use the format shown on page 29 in the textbook.
Question 3 (Bonus):

Now you want to write a program in your favorite language $\mathcal{L}$ that computes the square root function. Again, we only consider natural numbers, so that, for example, the square root of 17 is undefined.

**Hint:** If you create the series of perfect squares (0, 1, 4, 9, 16, …) and one of them matches the input, then you should be easily able to output its square root. Otherwise the program will just run forever.

a) Write down the program using macros. You are allowed to use the macros $V \leftarrow V'$, $V \leftarrow m$, GOTO $L$, and $W \leftarrow f(V_1, \ldots, V_n)$. Use at least one macro of the form $W \leftarrow f(V_1, \ldots, V_n)$, and whenever you do so, provide a program that computes $f(V_1, \ldots, V_n)$.

b) In your program, pick one of the macros of the form $W \leftarrow f(V_1, \ldots, V_n)$ and expand it using the convention given in Section 2.5 in the textbook. The expanded code can still contain the simple macros $V \leftarrow V'$, $V \leftarrow m$, or GOTO $L$; those do not need to be expanded any further. Write down the expanded version of your program.