Q1. On pages 55 and 56 of the textbook there is a proof for the correctness of the program `avg`. The author used the invariant

$$Inv(\mathtt{i}, \mathtt{sum})\colon 0 \leq \mathtt{i} < len(A) \wedge \mathtt{sum} = \sum_{k=0}^{\mathtt{i}-1} A[k].$$

As you can see, the invariant is a predicate in two variables: `i`, `sum`. These two variables are used in the program `avg`, but neither is the variable on which the induction proof is based. The author is using simple induction, but it is not very clear what the induction variable is (also the induction predicate itself is ambiguous). We want to make sure you understand what is going on there by having you re-write the proof by yourself in a style similar to the one we use in lectures. Here is the predicate you should be proving:

$$Q(j) : \text{At the beginning of the } j^{th} \text{ iteration}, \mathtt{sum} = \sum_{k=0}^{\mathtt{i}-1} A[k].$$

Remark 1: By the program's design, the variables $\mathtt{i}, \mathtt{sum}$ may change with each iteration (in other words, both are functions of j). This is why it might be more appropriate to write

$$Q(j) : \mathtt{sum}_j = \sum_{k=0}^{\mathtt{i}_j-1} A[k] \quad \text{or} \quad Q(j) : \mathtt{sum}(j) = \sum_{k=0}^{\mathtt{i}(j)-1} A[k].$$

where $\mathtt{i}_j$ (or $\mathtt{i}(j)$) means the value of program variable `i` at the beginning of the j^{th} iteration (the same with `sum`). That said, we believe that the version above Remark 1 is the best to work with as long as one understands that $\mathtt{i}, \mathtt{sum}$ are iteration dependent.

Remark 2: Using j as an index in Remark 1 has a different meaning from that which is intended by the author. The author is using indices to differentiate between the values of `i`, `sum` at the beginning of an (arbitrary) iteration and their values after the iteration is run. The end goal is to prove $Q(len(A))$. A proof by induction will show that

$$\forall j \in \{1, \ldots, len(A)\}, Q(j).$$

Your proof must follow the style used in lectures.

Proof. It will be shown using simple induction that $\forall j \in \{1, ..., \mathtt{len}(A)\}, Q(j)$.

Base Case. Let $j = 1$. At the beginning of the first iteration of the loop, $i = 0$ and $sum = 0$. Notice that in the expression $\sum_{k=0}^{-1} A[k]$, the lower bound is greater than the upper bound. This results in an empty sum that evaluates to 0. Therefore the base case holds.

Induction Hypothesis. Suppose that for some $j \in \{1, ..., \mathtt{len}(A)\}$, $Q(j)$ is true.

Induction Step. i increases by 1 at the end of each iteration, so at the beginning of the jth iteration, $i = j - 1 < \mathtt{len}(A)$, which means that the loop body is entered. From the induction hypothesis,

$$sum = \sum_{k=0}^{j-2} A[k]$$

Running through the loop, on line 9, $A[j-1]$ is added to sum. After this line is executed, the new value of sum becomes

$$sum = \sum_{k=0}^{j-2} A[k] + A[j-1] = \sum_{k=0}^{j-1} A[k]$$

Then, line 10 executes and the value of i is incremented and becomes j. After the line is executed, the program returns to the top of loop, signifying the beginning of the $j+1$th iteration. sum remains unchanged since the execution of line 9, and is equal to

$$\sum_{k=0}^{j-1} A[k] = \sum_{k=0}^{i-1} A[k]$$

as desired.

By the principle of simple induction, at the beginning of the jth iteration, $Q(j)$ holds for any $j \in \{1, 2, ..., \texttt{len}(A), \texttt{len}(A)+1\}$.

At the beginning of the $\texttt{len}(A)+1$th iteration, $i = \texttt{len}(A)$, so the loop terminates. The program returns $sum/\texttt{len}(A)$ on line 11, which is exactly the average of the numbers in A.

$\square$

Q2. –Following up on the Q1– We mentioned this in class, but it is good to remind you again of the fact that, proving $\forall j \in \{1, \ldots, len(A)\}, Q(j)$ is equivalent to proving $\forall n \in \mathbb{N}, Q'(n)$ where

$$Q'(n) : 0 \leq n < len(A) \Rightarrow Q(n+1).$$

Explain the equivalence.

Proof. It will be proven that the two statements imply each other, that is,

$$\forall j \in \{1, ..., \texttt{len}(A)\}, Q(j) \iff \forall n \in \mathbb{N}, 0 \leq n < \texttt{len}(A) \implies Q(n+1)$$

To prove the forward direction, suppose that $Q(j)$ holds true for all $j \in \{1, ..., \texttt{len}(A)\}$. Let n be a natural number such that $0 \leq n < \texttt{len}(A)$. It follows that $n \in \{0, 1, ..., \texttt{len}(A) - 1\}$, which implies that $n + 1 \in \{1, ..., \texttt{len}(A)\}$. By the original assumption, $Q(n + 1)$ is true.

Conversely, suppose that for all natural n, if $0 \leq n < \texttt{len}(A)$ then $Q(n + 1)$ is true. Let $j \in \{1, ..., \texttt{len}(A)\}$. Then $j - 1 \in \{0, ..., \texttt{len}(A) - 1\}$. It is clear that $j - 1$ is a natural number and that $0 \leq j < \texttt{len}(A)$. By the assumption in the beginning, $Q(j)$ is true.

Thus both statements are equivalent to each other.

□

Q3. Solve questions 6 to 10 on pages 64 to 66 of the textbook. There is a typo in Q8 line 12 (it should be `c = 1` instead of `c == 1`). Proofs must follow the lecture's format and level of clarity. Only **one** of the five questions will be selected for marking. A serious attempt of all the mentioned questions is a necessary condition for receiving more than 0 points.

Question 6.

1. Give a loop invariant that characterizes the values of a and y.

   Proof. Define the loop invariant to be

   $$Inv(x, y) : a = x + \frac{y(y+1)}{2} - 55$$

   It will be proven that at the beginning of the ith iteration of the loop, $Inv(x, y)$ is true.

   Base Case. At the beginning of the first iteration, $a = x$ and $y = 10$. Thus

   $$x + \frac{y(y+1)}{2} - 55 = x - \frac{10(11)}{2} - 55 = x - 55 + 55 = x = a$$

   The base case has been proven to be true.

   Induction Hypothesis. Now suppose that for some positive natural j, $Inv(a, y)$ holds true at the beginning of the jth iteration. y is initially 10 and is decremmented at the end of every iteration, so at the beginning of the jth iteration, $y = 10 - (j - 1) = 11 - j$. Then the loop invariant states that

   $$a = x + \frac{(11-j)(12-j)}{2} - 55$$

   Induction Step. If $a \leq 0$, the loop terminates and never reaches the beginning of the $j + 1$th iteration. Otherwise, on line 6 in the jth iteration, a is subtracted by $y = 11 - j$. Using the induction hypothesis, the value of a is now

   $$a = x + \frac{(11-j)(12-j)}{2} - 55 - (11-j) = x + \left(\frac{(11-j)(12-j)}{2} - (11-j)\right)$$

   $$= x + \frac{(11-j)(12-j-2)}{2} - 55 = x + \frac{(10-j)(11-j)}{2} - 55$$

   Finally, on line 7, the value of y is decremented by 1 and becomes $y = 10 - j$. The program then arrives at the beginning of the $j + 1$th iteration. Indeed, it is true that

   $$a = x + \frac{(10-j)(11-j)}{2} - 55 = x + \frac{y(y+1)}{2} - 55$$

   Thus $Inv(a, y)$ is true at the beginning of the $j + 1$th iteration. Verifying the validity of the loop invariant.

   □

2. Show that sometimes this code fails to terminate

Proof. Let $x = 56$. It wil be shown using simple induction that at the beginning of every iteration, $a > 0$, which implies that the loop can never terminate.

Consider the beginning of an arbitrary ith iteration. It is known that $y = 11 - j$. From the loop invariant found in part (a),

$$a = x + \frac{y(y+1)}{2} - 55 = 56 + \frac{(11-j)(12-j)}{2} - 55 = 1 + \frac{(11-j)(12-j)}{2}$$

Note that $\frac{(11-j)(12-j)}{2} < 0$ if and only if $11 < j$ and $j < 12$ at the same time, which is impossible for $j \in \mathbb{N}$. Thus $\frac{(11-j)(12-j)}{2} \geq 0$ so

$$a = 1 + \frac{(11-j)(12-j)}{2} \geq 1 > 0$$

Therefore, $a > 0$ at the beginning of all iterations, which means that the loop never terminates.

□

Question 7. State the pre- and post-conditions that the algorithms must satisfy, then prove that each algorithm is correct.

(a) *Proof.* Pre-condition: a is a non-zero number and b is a natural number.

Post-condition: Returns the value of a^b.

The correctness of the program will be proved using complete induction on b. For the entirety of the proof, a is fixed to be an arbitrary non-zero number.

Base Case. When $b = 0$, the method returns 1, which is exactly a^b.

Induction Hypothesis. Suppose that the program returns the correct value of a^b for all $b \leq n$, for some $n \in \mathbb{N}$.

Induction Step. When $b = n + 1$, $b \geq 1$, so line 3 is not reached. Consider two possible cases:

If b is even, line 5 and 6 will be executed. A new variable x is initialized to be equal to `exp_rec(a, b / 2)`. Since $\frac{b}{2}$ is natural and less than or equal to n, By the induction hypthesis, the value of x is $a^{\frac{b}{2}}$. The program returns `x * x`, which is equal to $a^{\frac{b}{2}} \cdot a^{\frac{b}{2}} = a^b$ on line 6.

If b is odd, line 8 and 9 will be executed. By a similar argument as the case above, a variable x is initialized to the value $a^{\frac{b-1}{2}}$ and `x * x * a` is returned, which has the value of $a^{\frac{b-1}{2}} \cdot a^{\frac{b-1}{2}} \cdot a = a^b$

Therefore, it can be concluded that the program satisfies the postconditions for all natural b.

□

(b) *Proof.* Define the loop invariant

$$P(\texttt{mult}, \texttt{exp})$$

□