1 Part 1

Markov Chain

We have two states :First item, Not first item.
The system is in "First item" when we have exactly one item in the current box that is being filled. The system is in "Not first item" when the current box contains more than one item.
X is the random variable for the weight of the current box.
W is the random variable for the weight of incoming item.

Transition matrix

The transition probabilities are as follows:
P_{FF} is the probability of going from state "First item" to state "First item". This happens when the weight of the new box (that has a single item) plus the incoming item exceeds the maximum allowed. Hence, we have a transition from one new box to another new box.
P_{NF} is the probability of going from state "Not first item" to state "First item". This happens when the weight of the current box plus the new item exceeds the maximum weight. Hence, we have a transition to a new box to accommodate the newest item.
P_{FN} is the probability of going from state "First item" to state "Not first item". This happens when the new box is able to accept the newest item and thereby becoming an "Not First item".
P_{NN} is the probability of going from state "Not first item" to state "Not first item". This happens when our current box (with 2 or more items) can accommodate the newest item while maintaining maximum weight. Therefore, there is not a need for a new box.

Mathematically, by conditioning on the item at time n we have:

$$\begin{aligned}
P_{FF} = P_{NF} \quad &= P(W > w_{max} - X) = \sum_{k=1}^{w_{max}} P(W > w_{max} - X | W = k) \cdot P(W = k) \\
P_{NN} = P_{FN} \quad &= P(W \leq w_{max} - X) = \sum_{k=1}^{w_{max}} P(W \leq w_{max} - X | W = k) \cdot P(W = k)
\end{aligned}$$

Now, $P(W \leq w_{max} - X | W = k)$ is the probability that the weight of the current box is less than or equal to $w_{max} - k$.

$$\begin{aligned}
P(W \leq w_{max} - X | W = k) &= P(X \leq w_{max} - k) \\
&= \sum_{i \leq w_{max} - k} P(X = i) \\
&= \sum_{i \leq w_{max} - k} \pi_i
\end{aligned}$$

Where π_i the stationary probability that the current box weighs w_i (calculated in Part 2). In other words, the probability that the current box weighs more than $w_{max} - k$ is the long run probability that box weighs 1 plus the long run probability that the box weighs 2, and so on up to $w_{max} - k$.

Similarly, we have

$$
\begin{aligned}
P(W > w_{max} - X | W = k) &= \sum_{i > w_{max} - k} P(X = i) \\
&= \sum_{i > w_{max} - k} \pi_i
\end{aligned}
$$

Putting everything together,

$$
\begin{aligned}
P_{FF} = P_{NF} &= \sum_{k=1}^{w_{max}} \left(\sum_{i > w_{max} - k} \pi_i \right) \cdot P(W = k) \\
P_{FN} = P_{NN} &= \sum_{k=1}^{w_{max}} \left(\sum_{i \leq w_{max} - k} \pi_i \right) \cdot P(W = k)
\end{aligned}
$$

Solution

Finally, the long run average number of items in any box is the average time for a system in state N to return to state N. This is $E(T_{FF})$, where T is the random variable for hitting times.
To find $E(T_{FF})$, we follow the similar argument in the book.
First, we condition on the first stop U.

$$E(T_{FF}) = P_{FN} \cdot E(T_{FF} | U = N) + P_{FF} \cdot E(T_{FF} | U = F)$$

where,

$$
\begin{aligned}
E(T_{FF} | U = F) &= 1 \\
E(T_{FF} | U = N) &= 1 + E(T_{NF})
\end{aligned}
$$

Simplifying the above, we arrive at

$$E(T_{FF}) = 1 + P_{FN} \cdot E(T_{NF})$$

By repeating the same argument for $E(T_{NF})$, we find

$$E(T_{NF}) = 1 + P_{NN} \cdot E(T_{NF})$$

Hence,

$$E(T_{NF}) = \frac{1}{1 - P_{NN}}$$

Putting this back in our previous expression of $E(T_{FF})$

$$E(T_{FF}) = 1 + \frac{P_{FN}}{1 - P_{NN}}$$

2 Part 2

2.1 Transition Matrix

2.2 Solution

The long run mean weight per box will be a weighted sum of average time for each X_n. Luckily, this is the π_i. Therefore, the long run mean weight per box is:

$$\sum_{i=1}^{w_{max}} i * \pi_i$$

3 Part 3

$P(Q = i)$ is the probability that item i is the first item to go in any box. Therefore, we condition that a new box is created to get the probability item i is first item in the new box.

$$\begin{aligned}
P(Q = i) &= P(W = i|X_n > w_{max} - W) \\
&= \frac{P((W = i) \cap (X_n > w_{max} - W))}{P(X_n > w_{max} - W)} \\
&= \frac{P(X_n > w_{max} - w_i)}{P(X_n > w_{max} - W)} \\
&= \frac{\sum\limits_{j > w_{max} - i} \pi_j}{\sum\limits_{k=1}^{w_{max}} \left(\sum\limits_{j > w_{max} - k} \pi_j \right) \cdot P(W = k)}
\end{aligned}$$

4 Part 4

Given $W_{max} = 10$ and P(W = i) = ci/10, i = 1,2,...,10 with the sum equal to 1 you get c = 10/55 $\approx$ 0.18. Running the following code for the values Q4(TRUE,FALSE,FALSE), Q4(TRUE,TRUE,FALSE), and Q4(FALSE,FALSE,TRUE).

```
Q4 <- function(testQ1 = FALSE, testQ2 = FALSE, testQ3 = FALSE)
{
    vmax = 10
    values = c(1:vmax)
    prob = c(1:vmax)
    values_in_new_box = c(rep(0,length(values)))
    constant = vmax/sum(values)
    prob = constant*values/vmax
    current_weight = 0
    number_of_boxes = 1
    boxes_total_weight = 0
    items = 100000
    for(i in 1:items)
    {
        random_value = sample(values, 1, TRUE, prob)
        boxes_total_weight = boxes_total_weight + random_value

        current_weight = current_weight + random_value
        if( current_weight > vmax )
        {
            number_of_boxes = number_of_boxes + 1
            current_weight = random_value
            values_in_new_box[random_value] = values_in_new_box[random_value] + 1
        }
    }

    number_of_items_per_box = items/number_of_boxes
    average_weight_per_box = boxes_total_weight/number_of_boxes

    values_in_new_box = values_in_new_box/number_of_boxes

    if(testQ1 == TRUE)
    {
        return(number_of_items_per_box)
    }
    if(testQ2 == TRUE)
```

```
    {
        return(average_weight_per_box)
    }
    if(testQ3 == TRUE)
    {
        return(values_in_new_box)
    }
}
```

This code yields the following simulated values for 100,000 samples. For the average number of items per box we get 1.132131. For the average weight per box we get 7.933018. For the probability of each weight starting a box we get the following table.

i	1	2	3	4	5
P(Q = i)	0.004532167	0.017698112	0.035645494	0.059699971	0.083403204
i	6	7	8	9	10
P(Q = i)	0.111683927	0.136916767	0.160178114	0.184130617	0.206100297

Here is where we compare these results to our numerical answers from Q1, Q2, and Q3.