This lecture talks about the third programming assignment:

Who gets the cake?

This assignment is inspired by the book Concrete Mathematics. Please refer to Section 1.3 for the mathematical foundation of such a problem.

Imagine that you have a group of n people and only one slice of cake. How do you decide who gets the cake?

This is how will decide who gets the cake.

The people forms a circle and mark from 0, 1, 2, and so on. In this example, there are 8 people, marked as 0, 1, 2, 3, to 7. We start from zero because the index of an array in C starts from zero.

Then, a number k, greater than one, is selected.

Let’s consider this example, when n is 8 and k is 5. We start from the person whose index is zero and mark this person as one. Then, the second person is marked as two. The next person is marked as three.

We continue this process until marking a person five. This person’s index is four and this person is removed from this game.

After removing this person, we continue counting from 1, 2, 3. They correspond to the indices of 5, 6, and 7. Then, we need to wrap around to index 0. The counts are put into the inner side of the circle.

Index 0 is now counted as number 4. Index one is counted as number five and removed.

We restart from one. Count two. The person whose index is 4 was removed earlier and thus is not counted in this round. Number three is the person whose index is 5. The person whose index is 7 is removed.

This slide shows the third round. The person whose index is one was removed and thus skipped in this round. This round removes the person whose index is 6.

This slide uses a different representation showing how this game is played. In this slide, we represent the people with indexes 0, 1, 2, as an array.

The top row shows the indexes. The bottom row shows the count from 1, 2, 3, 4, to 5.

The first removed person has index 4.

Then, we continue counting. After reaching the end, we wrap around to the beginning. The person that is removed the second has index one.

Continue counting, index 7 is removed.

The next round removes index 6.

.

.

The next round removes index 0.

.

The left side of this slide shows an execution of the program using 8 and 5 as the inputs. The program prints the indexes of the people that are removed.

As you can see, the first removed person has index 4, then 1, then 7, then 6.

Let’s see the program. This is the main function.

The program needs two arguments using A R G Vee one and A R G Vee two for the values of n and k.

The program first checks whether A R G C is three. If A R G C is not tree, the program stops by returning exit failure.

Then, the program uses S T R T O L. . to convert A R G Vee one and two to integers for the values of n and k.

If either value is one or less, the program stops because this program needs both n and k to be greater than one.

At line 24, the program calls the eliminate function.

.

At the top of the eliminate function, an array of integer is allocated. Each element is an integer.

You decide how to use the array.

The program will find the indexes of the removed people and print their indexes.

At the end of this function, the memory is released.

Please fill your code between lines 23 and 27, lines 30 and 35, as well as lines 35 to 40. Use as many lines as you need.

The make file has three test cases using 6 and 3, 6 and 4, 25 and 7. The expected outputs are also given to you.

You can see the expected output of the programs here.