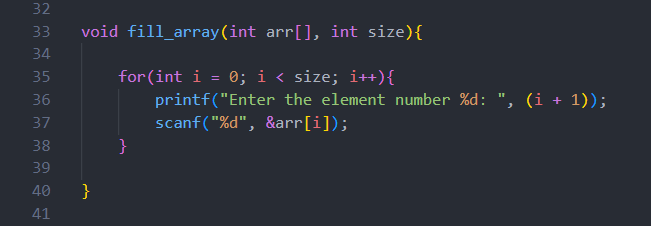
Problem Time: 30 min

Snapshots:

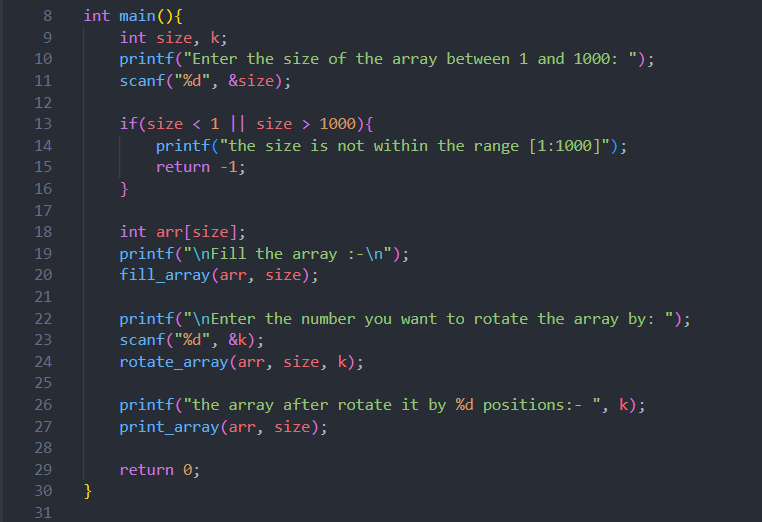
A computer screen shot of a program code

Description automatically generatedfill\_array function takes 2 parameters the array itself and the size of the array and asks the user to input each element in this array until he fills it.

rotate\_array function takes 3 parameters the array itself and the size of the array and the k positions you want to rotate the array by. inside the functions first, we calculate k % size to avoid the cases where k is larger than size then if the rotations variable is 0 that means the array will be the original one without any change so just use the return statement to terminate the function otherwise we will make nested loops where the outer one check is the variable rotations reach 0 or not to make (k % size) rotations and inside it we keep the last element in a variable and the inner loop shift all values of the array by one to the right and finally we assign the last element to the first index in the array to shift it as well A computer screen shot of a program code

Description automatically generatedthen decrement the rotations variable and continue until it becomes 0.

print\_array function takes 2 parameters the array itself and the size of the array and just simply prints the array inside square brackets and separates each element with a comma Except the last one.



in the main function, we ask the user for the size of the array between 0 and 1000 and if the size is not within that range, we terminate the program with a message to the user. otherwise, we will create an array with that size and use the fill\_array function to fill the array and then ask the user for the number he wants to rotate the array by then rotate the array using the rotate\_array function and finally print the array using the print\_array function.

A screenshot of a computer program

Description automatically generatedTest cases:

in case the size of the array is 5

A screen shot of a computer

Description automatically generated

in case the size is not within the range (for ex. 1500)

Algorithm:

Step 1: Read the array size from the user

Step 2: Check if the array size within the range or not and if not just terminate the program with a message to the user telling him that issue

Step 3: Create array with that size and fill the array using fill\_array function

Step 4: Read the k position that the user wants to rotate the array by and rotate it using rotate\_array function

Step 5: Print the array after the rotation using print\_array function