QUESTION

Rotate an Array Right by K Positions

Write a program to rotate an array right by k positions without using any built-in array or rotation

functions. For example, rotating [1, 2, 3, 4, 5] by 2 would give [4, 5, 1, 2, 3].

Instructions: You should implement the logic manually for rotating the array.

EXPLANATION/CODING

To begin our coding or implementation, first we need to understand the questions meaning then we will get to writing the logic for it. After that step we can go ahead with the coding part and implementation of it. The question focuses on writing a code that will manually rotate the array to right by the number of position given by the user, without using bultin array rotation functions.

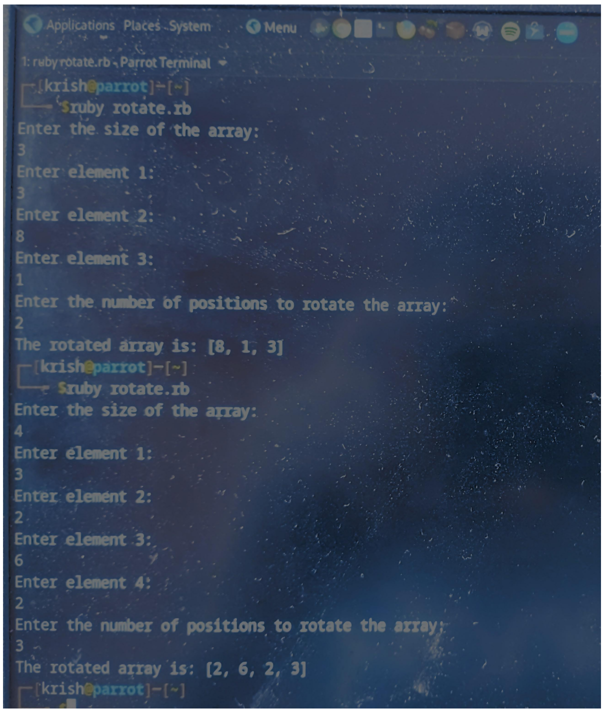
LOGIC USED

The purpose of the code is to rearrange an array by moving its elements to the right by a number of positions. Initially it checks if the number of rotations exceeds the length of the array. Then it iteratively saves the element of the array shifts all elements to the right, by one position and puts the saved last element at the beginning. This process repeats for the number of rotations to ensure that the array is rotated correctly without relying on predefined functions.

CODING/PROGRAM

```
def rotate_array(arr, k)
 n = arr.length
 k = k % n

 k.times do
  last_element = arr[-1]
  s = n - 1
  while s > 0
   arr[s] = arr[s - 1]
   s -= 1
  end
  arr[0] = last_element
 end

 arr
```

```
end


puts "Enter the size of the array:"

n = gets.to_i

arr = []


n.times do |i|

  puts "Enter element #{i + 1}:"

  element = gets.to_i

  arr << element

end


puts "Enter the number of positions to rotate the array:"

k = gets.to_i


result = rotate_array(arr, k)

puts "The rotated array is: #{result.inspect}"
```


Applications Places System
Menu
1: ruby rotate.rb - Parrot Terminal

```
┌─[krish@parrot]─[~]
└──╼ $ruby rotate.rb
Enter the size of the array:
3
Enter element 1:
3
Enter element 2:
8
Enter element 3:
1
Enter the number of positions to rotate the array:
2
The rotated array is: [8, 1, 3]
┌─[krish@parrot]─[~]
└──╼ $ruby rotate.rb
Enter the size of the array:
4
Enter element 1:
3
Enter element 2:
2
Enter element 3:
6
Enter element 4:
2
Enter the number of positions to rotate the array:
3
The rotated array is: [2, 6, 2, 3]
┌─[krish@parrot]─[~]
```

The process begins with the rotate_array function that shifts an array, to the right by a specified number of positions. First figuring out the arrays length. Adjusting the rotations to fit within the arrays size is crucial. Next comes a loop that saves the element of the array and shifts all elements one position to the right. Then follows placing the saved element at the front of the array after shifting is complete. This procedure continues for the number of rotations by adjusting the array without using the built in rotation functions in Ruby.

In the phase of the software operation focuses on engaging with users by requesting them to specify the arrays size and then proceeding to gather individual elements while also inquiring about the desired number of rotations, for the array. Once all necessary information is collected from the user interactions mentioned above the program then executes the rotate_array function to display the rotated array.

RUNNING THE CODE

To run the code we are going to use the Linux operating system, hence we need to download the ruby for it by using the commad “**sudo apt install ruby-full”.** Then we need to write the code in a text editor , here we use nano for it and save it as rotate.rb. And to run the code we just need to locate the file location using the terminal and execute it by using the command ruby rotate.rb.