Question 1

An array is a data structure that keeps items of the same kind in a single memory block. Each memory position in an array, P, of size Q, has a unique index, I (where 0=i=Q), which may be referred as P[i] or Pi. Reverse an integer array.

Example
P=[4,5,6]
Return [6,5,4]

Reverse an integer array. Description of the Function

complete the reverseArray function in any programming language or your choice or you may write sudo code for the same

The parameter(s) for reverseArray are:
int P [q]: the array to reverse

Returns
Int[q]: the reversed array

Input format
The first line includes an integer, Q, which represents the number of integers in P.
P is made up of Q space-separated numbers on the second line.

Constraints

- $1<=Q<=10^3$
- $1<=p[i]<=10^4$ where p[i] is the i^{th} integer in P

Sample input

5	8	7	6

Sample output

6	7	8	5

Complete the function in any programming language or your choice or you may write sudo code for the same

Question 2

This is a good way to get some practise with traversing a linked list. Print each node's data element, one per line, given a pointer to the head node of a linked list. There is nothing to output if the head reference is null (meaning the list is empty).

Create a function to print LinkedLists.

The parameter(s) for printLinkedList are as follows:
SinglyLinkedList The head of the list is referred to as the node head.
Print the value of each node in a new line

Input format
The first line of input contains p, which is the number of linked list entries. The data values for each node are contained in the next p lines, each with one element.

Complete the printlinkedlist in any programming language or your choice or you may write sudo code for the same

Constraints
1<=P<=1000
1<=list[i]<=1000 where list [i] is the i^{th} element of the linked list

Complete the function in any programming language or your choice or you may write sudo code for the same

Complete the function in any programming language or your choice or you may write sudo code for the same