1. **Python Program to reverse a given number.**

*Problem Solution*

Step 1: Take the value of the integer and store in a variable.  
Step 2: Using a while loop, get each digit of the number and store the reversed number in another variable.  
Step 3: Print the reverse of the number.  
Step 4: Exit

Program/Source Code

n=int(input("Enter number: "))

rev=0

while(n>0):

dig=n%10

rev=rev\*10+dig

n=n//10

print("Reverse of the number:",rev)

*Program Explanation*

1. User must first enter the value and store it in a variable n.  
2. The while loop is used and the last digit of the number is obtained by using the modulus operator.  
3. The last digit is then stored at the one’s place, second last at the ten’s place and so on.  
4. The last digit is then removed by truly dividing the number with 10.  
5. This loop terminates when the value of the number is 0.  
6. The reverse of the number is then printed.

*Runtime Test Cases*

Case 1:

Enter number: 124

Reverse of the number: 421

Case 2:

Enter number: 4538

Reverse of the number: 8354

1. **Python Program to exchange the values of two numbers without using a temporary variable.**

*Problem Description*

The program takes both the values from the user and swaps them without using temporary variable.

*Problem Solution*

1. Take the values of both the elements from the user.  
2. Store the values in separate variables.  
3. Add both the variables and store it in the first variable.  
4. Subtract the second variable from the first and store it in the second variable.  
5. Then, subtract the first variable from the second variable and store it in the first variable.  
6. Print the swapped values.  
7. Exit.

*Program/Source Code*

Here is source code of the Python Program to exchange the values of two numbers without using a temporary variable. The program output is also shown below.

a=int(input("Enter value of first variable: "))

b=int(input("Enter value of second variable: "))

a=a+b

b=a-b

a=a-b

print("a is:",a," b is:",b)

*Program Explanation*

1. User must first enter the values for both the elements.  
2. The first element is assigned the sum of the first two elements.  
3. Second element is assigned the difference between the sum in the first variable and the second variable, which is basically the first element.  
4. Later the first element is assigned the difference between the sum in the variable and the second variable, which is the second element.  
5. Then the swapped values are printed.

Runtime Test Cases

Case 1

Enter value of first variable: 3

Enter value of second variable: 5

a is: 5 b is: 3

Case 2

Enter value of first variable: 56

Enter value of second variable: 25

a is: 25 b is: 56

1. **Python Program to read a number n and compute n+nn+nnn.**

Problem Description

The program takes a number n and computes n+nn+nnn.

Problem Solution

1. Take the value of a element and store in a variable n.  
2. Convert the integer into string and store it in another variable.  
3. Add the string twice so the string gets concatenated and store it in another variable.  
4. Then add the string thrice and assign the value to the third variable.  
5. Convert the strings in the second and third variables into integers.  
6. Add the values in all the integers.  
7. Print the total value of the expression.  
8. Exit.

Program/Source Code

Here is the source code of the Python Program to read a number n and compute n+nn+nnn. The program output is also shown below.

n=int(input("Enter a number n: "))

temp=str(n)

t1=temp+temp

t2=temp+temp+temp

comp=n+int(t1)+int(t2)

print("The value is:",comp)

Program Explanation

1. User must first enter the value and store it in a variable n.  
2. The integer is converted to string for concatenation of the value of n.  
3. The string is then concatenated once and twice and stored in separate variables.  
4. Later to find the total sum, the string is converted back to integer.  
5. The total value of the expression is then printed.

Runtime Test Cases

Case 1:

Enter a number n: 5

The value is: 615

Case 2:

Enter a number n: 20

The value is: 204060

1. **Python Program to check whether a given number is a palindrome.**

*Problem Description*

The program takes a number and checks whether it is a palindrome or not.

Problem Solution

1. Take the value of the integer and store in a variable.  
2. Transfer the value of the integer into another temporary variable.  
3. Using a while loop, get each digit of the number and store the reversed number in another variable.  
4. Check if the reverse of the number is equal to the one in the temporary variable.  
5. Print the final result.  
6. Exit.

*Program/Source Code*

Here is source code of the Python Program to check whether a given number is a palindrome. The program output is also shown below.

n=int(input("Enter number:"))

temp=n

rev=0

while(n>0):

dig=n%10

rev=rev\*10+dig

n=n//10

if(temp==rev):

print("The number is a palindrome!")

else:

print("The number isn't a palindrome!")

*Program Explanation*

1. User must first enter the value of the integer and store it in a variable.  
2. The value of the integer is then stored in another temporary variable.  
3. The while loop is used and the last digit of the number is obtained by using the modulus operator.  
4. The last digit is then stored at the one’s place, second last at the ten’s place and so on.  
5. The last digit is then removed by truly dividing the number with 10.  
6. This loop terminates when the value of the number is 0.  
7. The reverse of the number is then compared with the integer value stored in the temporary variable.  
8. If both are equal, the number is a palindrome.  
9. If both aren’t equal, the number isn’t a palindrome.  
10. The final result is then printed.

*Runtime Test Cases*

Case 1

Enter number:121

The number is a palindrome!

Case 2

Enter number:567

The number isn't a palindrome!

1. **Python Program to read a number n and print an identity matrix of the desired size.**

*Problem Description*

The program takes a number n and prints an identity matrix of the desired size.

Problem Solution

1. Take a value from the user and store it in a variable n.  
2. Use two for loop where the value of j ranges between the values of 0 and n-1 and value of i also ranges between 0 and n-1.  
3. Print the value of 1 when i is equal to j and 0 otherwise.  
4. Exit.

*Program/Source Code*

Here is the source code of the Python Program to read a number n and print an identity matrix of the desired size. The program output is also shown below.

n=int(input("Enter a number: "))

for i in range(0,n):

for j in range(0,n):

if(i==j):

print("1",sep=" ",end=" ")

else:

print("0",sep=" ",end=" ")

print()

*Program Explanation*

1. User must first enter the value and store it in a variable n.  
2. The outer for loop enables j to range between 1 and n-1 (as n is not included) while the inner for loop also enables i to range between 1 and n-1.  
3. For each iteration, the value of 1 is printed if i is equal to j and value of 0 is printed otherwise.  
4. The sep and end parameters of the print function help in formatting and print() allows values to printed in a new line for each iteration of the outer for loop.

Runtime Test Cases

Case 1:

Enter a number: 4

1 0 0 0

0 1 0 0

0 0 1 0

0 0 0 1

Case 2:

Enter a number: 5

1 0 0 0 0

0 1 0 0 0

0 0 1 0 0

0 0 0 1 0

0 0 0 0 1

1. **Python Program to read a number n and print an inverted star pattern of the desired size.**

*Problem Description*

The program takes a number n and prints an inverted star pattern of the desired size.

Problem Solution

1. Take a value from the user and store it in a variable n.  
2. Use a for loop where the value of i ranges between the values of n-1 and 0 with a decrement of 1 with each iteration.  
3. Multiply empty spaces with n-i and ‘\*’ with i and print both of them.  
4. Exit.

*Program/Source Code*

Here is the source code of the Python Program to read a number n and print an inverted star pattern of the desired size. The program output is also shown below.

n=int(input("Enter number of rows: "))

for i in range (n,0,-1):

print((n-i) \* ' ' + i \* '\*')

*Program Explanation*

1. User must first enter the value and store it in a variable n.  
2. The for loop enables i to range between n-1 and 0 with a decrement of 1 with each iteration.  
3. For each iteration, ” ” is multiplied with n-i and ‘\*’ is multiplied with i to ensure correct spacing of the stars.  
4. The required pattern is printed.

Runtime Test Cases

Case 1:

Enter number of rows: 5

\*\*\*\*\*

\*\*\*\*

\*\*\*

\*\*

\*

Case 2:

Enter number of rows: 10

\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*

\*\*\*

\*\*

\*

1. **Python Program to read two numbers and print their quotient and remainder.**

*Problem Description*

The program takes two numbers and prints the quotient and remainder.

Problem Solution

1. Take in the first and second number and store it in separate variables.  
2. Then obtain the quotient using division and the remainder using modulus operator.  
3. Exit.

*Program/Source Code*

Here is the source code of the Python Program to read two numbers and print their quotient and remainder. The program output is also shown below.

a=int(input("Enter the first number: "))

b=int(input("Enter the second number: "))

quotient=a//b

remainder=a%b

print("Quotient is:",quotient)

print("Remainder is:",remainder)

*Program Explanation*

1. User must enter the first and second number .  
2. The quotient is obtained using true division (// operator).  
3. The modulus operator gives the remainder when a is divided by b.

Runtime Test Cases

Case 1:

Enter the first number: 15

Enter the second number: 7

Quotient is: 2

Remainder is: 1

Case 2:

Enter the first number: 125

Enter the second number: 7

Quotient is: 17

Remainder is: 6

1. **Python Program to accept three distinct digits and print all possible combinations from the digits.**

*Problem Description*

The program takes three distinct numbers and prints all possible combinations from the digits.

*Problem Solution*

1. Take in the first, second and third number and store it in separate variables.  
2. Then append all the three numbers to the list.  
3. Use three for loops and print the digits in the list if none of their indexes are equal to each other.  
4. Exit.

*Program/Source Code*

Here is the source code of the Python Program to accept three distinct digits and prints all possible combinations from the digits. The program output is also shown below.

a=int(input("Enter first number:"))

b=int(input("Enter second number:"))

c=int(input("Enter third number:"))

d=[]

d.append(a)

d.append(b)

d.append(c)

for i in range(0,3):

for j in range(0,3):

for k in range(0,3):

if(i!=j&j!=k&k!=i):

print(d[i],d[j],d[k])

*Program Explanation*

1. User must enter the first, second and third number.  
2. All the elements are appending into a list for the ease of comparison.  
3. The for loops range from 0-2 which are basically the indexes of the three elements in the list.  
4. If none of the indexes are equal to each other, the element associated with the particular element in the list is printed.

Runtime Test Cases

Case 1:

Enter first number:1

Enter second number:2

Enter third number:3

1 2 3

1 3 2

2 1 3

2 3 1

3 1 2

3 2 1

Case 2:

Enter first number:5

Enter second number:7

Enter third number:3

5 7 3

5 3 7

7 5 3

7 3 5

3 5 7

3 7 5

1. **Python Program to read a number n and print and compute the series “1+2+…+n=”.**

*Problem Description*

The program takes a number n and prints and computes the series “1+2+…+n=”.

Problem Solution

1. Take a value from the user and store it in a variable n.  
2. Use a for loop where the value of i ranges between the values of 1 and n.  
3. Print the value of i and ‘+’ operator while appending the value of i to a list.  
4. Then find the sum of elements in the list.  
5. Print ‘=’ followed by the total sum.  
6. Exit.

*Program/Source Code*

Here is the source code of the Python Program to read a number n and print and compute the series “1+2+…+n=”. The program output is also shown below.

n=int(input("Enter a number: "))

a=[]

for i in range(1,n+1):

print(i,sep=" ",end=" ")

if(i<n):

print("+",sep=" ",end=" ")

a.append(i)

print("=",sum(a))

print()

*Program Explanation*

1. User must first enter the value and store it in a variable n.  
2. The for loop enables i to range between 1 and n (as n+1 is not included).  
3. For each iteration, the value of i is printed.  
4. ‘+’ operator is printed only if i<="" div="">

Runtime Test Cases

Case 1:

Enter a number: 4

1 + 2 + 3 + 4 = 10

Case 2:

Enter a number: 5

1 + 2 + 3 + 4 + 5 = 15

1. **Python Program to compute a polynomial equation given that the coefficients of the polynomial are stored in the list.**

*Problem Description*

The program takes the coefficients of the polynomial equation and the value of x and gives the value of the polynomial.

*Problem Solution*

1. Import the math module.  
2. Take in the coefficients of the polynomial equation and store it in a list.  
3. Take in the value of x.  
4. Use a for loop and while loop to compute the value of the polynomial expression for the first three terms and store it in a sum variable.  
5. Add the fourth term to the sum variable.  
6. Print the computed value.  
7. Exit.

*Program/Source Code*

Here is source code of the Python Program to compute a polynomial equation given that the coefficients of the polynomial are stored in a list. The program output is also shown below.

import math

print("Enter the coefficients of the form ax^3 + bx^2 + cx + d")

lst=[]

for i in range(0,4):

a=int(input("Enter coefficient:"))

lst.append(a)

x=int(input("Enter the value of x:"))

sum1=0

j=3

for i in range(0,3):

while(j>0):

sum1=sum1+(lst[i]\*math.pow(x,j))

break

j=j-1

sum1=sum1+lst[3]

print("The value of the polynomial is:",sum1)

*Program Explanation*

1. The math module is imported.  
2. User must enter the coefficients of the polynomial which is stored in a list.  
3. User must also enter the value of x.  
4. The value of i ranges from 0 to 2 using the for loop which is used to access the coefficients in the list.  
5. The value of j ranges from 3 to 1, which is used to determine the power for the value of x.  
6. The value of the first three terms is computed this way.  
7. The last term is added to the final sum.  
8. The final computed value is printed.

*Runtime Test Cases*

Case 1:

Enter the coefficients of the form ax^3 + bx^2 + cx + d

Enter coefficient:3

Enter coefficient:4

Enter coefficient:5

Enter coefficient:6

Enter the value of x:2

The value of the polynomial is: 56.0

Case 2:

Enter the coefficients of the form ax^3 + bx^2 + cx + d

Enter coefficient:2

Enter coefficient:5

Enter coefficient:6

Enter coefficient:3

Enter the value of x:1

The value of the polynomial is: 16.0

**Program to Solve by yourself**

1. Write a Python Program to Find the Factorial of a Number

# Write a Python Program to Print the Fibonacci sequence