class Node:

def \_\_init\_\_(self, coeff, power):

self.coeff = coeff

self.power = power

self.next = None

def insert\_term(head, coeff, power):

new\_node = Node(coeff, power)

if head is None or power > head.power:

new\_node.next = head

return new\_node

temp = head

while temp.next and temp.next.power >= power:

if temp.next.power == power:

temp.next.coeff += coeff

return head

temp = temp.next

if temp.power == power:

temp.coeff += coeff

else:

new\_node.next = temp.next

temp.next = new\_node

return head

def add\_poly(p1, p2):

result = None

while p1 and p2:

if p1.power == p2.power:

result = insert\_term(result, p1.coeff + p2.coeff, p1.power)

p1 = p1.next

p2 = p2.next

elif p1.power > p2.power:

result = insert\_term(result, p1.coeff, p1.power)

p1 = p1.next

else:

result = insert\_term(result, p2.coeff, p2.power)

p2 = p2.next

while p1:

result = insert\_term(result, p1.coeff, p1.power)

p1 = p1.next

while p2:

result = insert\_term(result, p2.coeff, p2.power)

p2 = p2.next

return result

def display\_poly(head):

if head is None:

print("0")

return

temp = head

while temp:

print(f"{temp.coeff}x^{temp.power}", end=" ")

if temp.next:

print("+", end=" ")

temp = temp.next

print()

def get\_polynomial():

head = None

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

for \_ in range(n):

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

power = int(input("Enter power: "))

head = insert\_term(head, coeff, power)

return head

print("Enter first polynomial:")

poly1 = get\_polynomial()

print("Enter second polynomial:")

poly2 = get\_polynomial()

print("\nFirst Polynomial:")

display\_poly(poly1)

print("Second Polynomial:")

display\_poly(poly2)

sum\_poly = add\_poly(poly1, poly2)

print("Sum of Polynomials:")

display\_poly(sum\_poly)

**OUTPUT:**



