SCHOOL OF OPEN LEARNING

UNIVERSITY OF DELHI

PROGRAM:- C++

COURSE:- B.A. PROGRAMME WITH COMPUTER APPLICATIONS

SEMESTER:- 1

NAME :- Pankaj Ahlawat

SOL ROLL NO. :-24-1-11-000076

Index

S.No.	Experiment	Date	Sign
1	Write a program to swap two numbers without using another variable.		
2	Write a program to concatenate two strings.		
3	Write a program to calculate the area of a circle.		
4	Write a program to calculate the square of a number.		
5	Write a program to print the value of π (pi).		
6	Write a program to take input from the user and use arithmetic operators.		
7	Write a program that calculates the circumference of a circle based on the radius entered by the user.		
8	Write a program to take marks of five subjects from the user, find the total and percentage, and print the grade.		
9	Write a program to take three sides of a triangle and determine whether it is scalene, isosceles, equilateral, or a right triangle.		
10	Write a program that accepts the user's first and last name and prints them in reverse order with a space between them.		
11	Write a program to check whether a number is even or odd.		
12	Write a program to check whether a number is divisible by 13 or not.		
13	Write a program to print 100 multiples of 17.		
14	Write a program to print 50 multiples of 21.		

15	Write a program to print the squares of the first 20 numbers.		
16	Write a program to print the cubes of the first 30 numbers.		
17	Write a program to print the patterns.		
18	Write a program to find the sum of the series: 1 + 2 + 3 + 4 + ... + n.		
19	Write a program to find the sum of the series: 1/1! + 2/2! + 3/3! + 4/4! + ... + n/n!.		
20	Write a program to find the sum of the series: 1 - 2 + 3 - 4 + 5 - 6 + ... + n.		
21	Create a program that allows the user to guess a secret number between 1 and 100. The program should keep prompting the user until they guess the correct number.		
22	Write a program to generate the Fibonacci sequence up to a given number of terms.		
23	Create a program that allows two players to play a game of rock, paper, scissors.		
24	Write a program that opens a file and handles a FileNotFoundError exception if the file does not exist.		
25	Write a program that executes an operation on a list and handles an IndexError exception if the index is out of range.		
26	Write a program that performs division and handles an ArithmeticError exception if there is a division-related error.		
27	Write a program to print the reverse number pattern using a for loop.		
28	Write a program to print all prime numbers within a given range.		
29	Write a program to reverse the digits of a number.		

----------****----------

Experiment 1: Write a program to swap two numbers without using another variable.

Program:

```
a= 10
b = 20

print("\na = ",a," b= ", b)
a , b = b, a
print("a = ",a," b= ", b, "\n")
```

Output:

```
a =  10  b=  20
a =  20  b=  10
```

----------****----------

Experiment 2: Write a program to concatenate two strings..

Program:

```
str1 = "Hello"
str2 = "World!"
str3 = str1 + str2
print(str3)

str4 = "Nice to"
str5 = "Meet Ya!"
str6 = str4 + str5
print(str6)
```

```
print("    ")
```

Output:

HelloWorld!
Nice toMeet Ya!

----------****----------

Experiment 3: Write a program to calculate the area of a circle.

Program:

```
import math
radius = 7
area = math.pi * radius * radius
print("Area of Circle = ", area)
```

Output:

Area of Circle = 153.93804002589985

----------****----------

Experiment 4: Write a program to calculate the square of a number.

Program:

```
num = int(input("Enter a number: "))
```

```
square = num ** 2
print(f"The square of {num} is {square}\n")
```

Output:

```
Enter a number: 56
The square of 56 is 3136
```

----------****----------

Experiment 5: Write a program to print the value of π (pi).

Program:

```
import math
print("\nOutput\n")
print(math.pi)
print("   ")
```

Output:

```
3.141592653589793
```

----------****----------

Experiment 6: Write a program to take input from the user and use arithmetic operators.

Program:

```
num1 = int(input("Enter first number: "))
num2 = int(input("Enter second number: "))
```

```
operator = input("Enter operator (+, -, *, /): ")

if operator == "+":
    result = num1 + num2
elif operator == "-":
    result = num1 - num2
elif operator == "*":
    result = num1 * num2
else:
    result = num1 / num2

print("The result is", result)
```

Output:

```
Enter first number: 6
Enter second number: 7
Enter operator (+, -, *, /): *
The result is 42
```

----------****----------

Experiment 7: Write a program that calculates the circumference of a circle based on the radius entered by the user.

Program:

```
import math

radius = float(input("Enter radius: "))
```

```
circumference = 2 * math.pi * radius
print("The circumference is", circumference)
```

Output:

```
Enter radius: 5
The circumference is 31.41592653589793
```

----------****----------

Experiment 8: Write a program to take marks of five subjects from the user, find the total and percentage, and print the grade.

Program:

```
eng = float(input("Score in English: "))
cs = float(input("Score in Computer Science: "))
maths = float(input("Score in Mathematics: "))
phy = float(input("Score in Physics: "))
business = float(input("Score in Business Studies: "))

marks = [eng, cs, maths, phy, business]
total_marks = sum(marks)
percentage = total_marks / 500 * 100
grade = ""

if percentage >= 90:
    grade = "A+"
elif 80 <= percentage < 90:
    grade = "A"
```

```
elif 70 <= percentage < 80:
    grade = "B+"
elif 60 <= percentage < 70:
    grade = "B"
elif 50 <= percentage < 60:
    grade = "C"
elif 40 <= percentage < 50:
    grade = "D"
else:
    grade = "F"

print("Your total marks are:", total_marks)
print("Your percentage is:", percentage)
print("Your grade is:", grade)
```

Output:

```
Score in English: 75
Score in Computer Science: 80
Score in Mathematics: 90
Score in Physics: 65
Score in Business Studies: 85
Your total marks are: 395.0
Your percentage is: 79.0
Your grade is: B+
```

----------****----------

Experiment 9: Write a program to take three sides of a triangle and determine whether it is scalene, isosceles, equilateral, or a right triangle.

Program:

```
sides = [float(input("Enter the first side of the triangle: ")),
        float(input("Enter the second side of the triangle: ")),
        float(input("Enter the third side of the triangle: "))]

if all(x == sides[0] for x in sides):
    print("Equilateral triangle")
elif sides[0] == sides[1] or sides[1] == sides[2] or sides[2] ==
sides[0]:
    print("Isosceles triangle")
else:
    sides.sort()
    if (sides[2]**2 == sides[0]**2 + sides[1]**2):
        print("Right triangle")
    else:
        print("Its none of them")
```

Output:

```
Enter the first side of the triangle: 5
Enter the second side of the triangle: 9
Enter the third side of the triangle: 9
Isosceles triangle
```

----------****----------

Experiment 10: Write a program that accepts the user's first and last name and prints them in reverse order with a space between them.

Program:

```
name = input("Enter your full name: ")
last_name, first_name = name.split()
print(first_name + " " + last_name)
```

Output:

```
Enter your full name: Pankaj Ahlawat
Ahlawat Pankaj
```

----------****----------

Experiment 11: Write a program to check whether a number is even or odd.

Program:

```
num = int(input("Enter a number: "))
if num % 2 == 0:
    print("Even")
else:
    print("Odd")
```

Output:

```
Enter a number: 55
Odd
```

----------****----------

Experiment 12: Write a program to check whether a number is divisible by 13.

Program:

```
num = int(input("Enter a number: "))
if num % 13 == 0:
    print("Divisible")
else:
    print("Not divisible")
```

Output:

```
Enter a number: 69
Not divisible
```

----------****----------

Experiment 13: Write a program to print 100 multiples of 17.

Program:

```
for i in range(1, 101):
    print(i*17)
```

Output:

```
17
34
51
```

68
85
102
119
136
153
170
187
204
221
238
255
272
289
306
323
340
357
374
391
408
425
442
459
476
493
510
527
544
561

578
595
612
629
646
663
680
697
714
731
748
765
782
799
816
833
850
867
884
901
918
935
952
969
986
1003
1020
1037
1054
1071

1088
1105
1122
1139
1156
1173
1190
1207
1224
1241
1258
1275
1292
1309
1326
1343
1360
1377
1394
1411
1428
1445
1462
1479
1496
1513
1530
1547
1564
1581

1598
1615
1632
1649
1666
1683
1700

----------****----------

Experiment 14: Write a program to print 50 multiples of 21.

Program:

```
for i in range(1, 50):
    print(i*21)
```

Output:

21
42
63
84
105
126
147
168
189
210
231
252

273
294
315
336
357
378
399
420
441
462
483
504
525
546
567
588
609
630
651
672
693
714
735
756
777
798
819
840
861
882

903
924
945
966
987
1008
1029

----------****----------

Experiment 15: Write a program to print the squares of the first 20 numbers.

Program:

```
for i in range(1, 21):
    print(i**2)
```

Output:

1
4
9
16
25
36
49
64
81
100
121
144

169
196
225
256
289
324
361
400

----------****----------

Experiment 16: Write a program to print the cubes of the first 30 numbers.

Program:

```
for i in range(1, 31):
    print(i**3)
```

Output:

1
8
27
64
125
216
343
512
729
1000

1331
1728
2197
2744
3375
4096
4913
5832
6859
8000
9261
10648
12167
13824
15625
17576
19683
21952
24389
2700

----------****----------

Experiment 17: Write a program to print the pattern.

Part A: Alphabet-Pyramid:

```
height = int(input("Enter height of alphabet pyramid: "))

for i in range(height):
    print(' ' * (height - i - 1), end='')
```

```
    for j in range(i + 1):
        print(chr(65 + j), end=' ')

    print()
```

Output:

```
Enter height of alphabet pyramid: 5
    A
   A B
  A B C
 A B C D
A B C D E
```

Part B: Hollow-Star-Pyramid:

```
height = int(input("Enter height of number pyramid: "))

for i in range(height):
    print(' ' * (height - i - 1), end='')

    if i == height - 1:
        print('*' * (2 * i + 1))
    else:
        print('*', end='')
        if i > 0:
            print(' ' * (2 * i - 1), end='')
            print('*')
        else:
            print()
```

Output:

```
Enter the height of the number pyramid: 5
    *
   * *
  *   *
 *     *
*********
```

Part C: Pyramid:

```
height = int(input("Enter height of pyramid: "))
width = 2*height - 1

for i in range(height):
    print(' '*(width-i), end='')
    print('*'*(2*i+1))
```

Output:

```
Enter height of pyramid: 5
        *
       ***
      *****
     *******
    *********
```

Part D: Inverted Pyramid:

```
height = int(input("Enter height of pyramid: "))
width = 2*height - 1
```

```
for i in range(height):
    print(' ' * i, end='')
    print('*' * (width - 2*i))
```

Output:

```
a =  10  b=  20
a =  20  b=  10
```

Part E: Number-Pyramid:

```
height = int(input("Enter height of number pyramid: "))

for i in range(1, height + 1):
    print(' ' * (height - i), end='')

    for j in range(1, i + 1):
        print(j, end=' ')

    print()
```

Output:

```
Enter height of number pyramid: 5
    1
   1 2
  1 2 3
 1 2 3 4
1 2 3 4 5
```

----------****----------

Experiment 18: Write a program to find the sum of the series: 1 + 2 + 3 + 4 + ... + n.

Program:

```
n = int(input("Enter n: "))
sum = 0
for i in range(1, n+1):
    sum += i
print("Sum of the series 1 + 2 + 3 + ... + 9 is:", sum)
```

Output:

```
Enter n: 50
Sum of the series 1 + 2 + 3 + ... + 50 is: 1275
```

----------****----------

Experiment 19: Write a program to find the sum of the series: 1/1! + 2/2! + 3/3! + 4/4! + ... + n/n!..

Program:

```
import math

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

sum_of_series = 0

for i in range(1, n + 1):
    sum_of_series += i / math.factorial(i)
```

```
print(f"The sum of the series 1/1! + 2/2! + ... + {n}/{n}! is:
{sum_of_series}")
```

Output:

```
Enter the value of n: 5
The sum of the series 1/1! + 2/2! + ... + 5/5! is:
2.708333333333333
```

----------****----------

Experiment 20: Write a program to find the sum of the series: 1 - 2 + 3 - 4 + 5 - 6 + ... + n.

Program:

```
n = int(input("Enter the value of n: "))
sum_of_series = 0

for i in range(1, n + 1):
    if i % 2 == 0:
        sum_of_series -= i
    else:
        sum_of_series += i

print(f"The sum of the series 1 - 2 + 3 - 4 + ... ± {n} is:
{sum_of_series}")
```

Output:

```
Enter the value of n: 10
```

The sum of the series 1 - 2 + 3 - 4 + ... ± 10 is: -5

----------****----------

Experiment 21: Create a program that allows the user to guess a secret number between 1 and 100. The program should keep prompting the user until they guess the correct number.

Program:

```
import random

secret_number = random.randint(1, 100)

num = 0
while num != secret_number:
    num = int(input("Enter your guess: "))
    if num < secret_number:
        print("guess higher")
    elif num > secret_number:
        print("guess lower")
    elif num == secret_number:
        print("YOU GUESSED IT!!")
```

Output:

```
Enter your guess: 50
guess higher
Enter your guess: 70
guess lower
Enter your guess: 60
```

```
guess higher
Enter your guess: 65
guess higher
Enter your guess: 66
YOU GUESSED IT!!
```

----------****----------

Experiment 22: Write a program to generate the Fibonacci sequence up to a given number of terms.

Program:

```
def fib(n):
    if n <= 0:
        return []
    elif n == 1:
        return [0]
    elif n == 2:
        return [0, 1]

    seq = [0, 1]
    for i in range(2, n):
        next_term = seq[-1] + seq[-2]
        seq.append(next_term)

    return seq

num_terms = int(input("Enter the number of terms for the
Fibonacci sequence: "))
```

```
if num_terms <= 0:
   print("Please enter a positive integer.")
else:
   fibonacci_sequence = fib(num_terms)
   print(f"Fibonacci sequence ({num_terms} terms):
{fibonacci_sequence}")
```

Output:

```
Enter the number of terms for the Fibonacci sequence: 10
Fibonacci sequence (10 terms): [0, 1, 1, 2, 3, 5, 8, 13, 21, 34]
```

----------****----------

Experiment 23:Create a program that allows two players to play a game of rock, paper, scissors

Program:

```
choices = ["rock", "paper", "scissors"]

def winner_is(p1_choice, p2_choice):
   if p1_choice == p2_choice :
      return "its a tie"
   elif (p1_choice == "rock" and p2_choice == "scissors") or
(p1_choice == "paper" and p2_choice == "scissors") or(p1_choice
== "scissors" and p2_choice == "paper"):
      return "p1 wins"
   else:
      return "p2 win"

print("Choices: rock, paper, scissors\n")
```

```
player1 = input("Player 1, enter your choice: ").strip().lower()
player2 = input("Player 2, enter your choice: ").strip().lower()

if player1 not in choices or player2 not in choices:
    print("Invalid input! Choices must be 'rock', 'paper', or
'scissors'.")
else:
    result = winner_is(player1, player2)
    print(result)
```

Output:

```
Choices: rock, paper, scissors

Player 1, enter your choice: rock
Player 2, enter your choice: paper
p2 win
```

----------****----------

Experiment 24: Write a program to handle division by zero, ValueError, and NameError using Try except else finally block.

Program:

```
def divide_numbers():
    try:
        numerator = float(input("Enter the numerator: "))
        denominator = float(input("Enter the denominator: "))
```

```
        result = numerator / denominator

    except ZeroDivisionError:
        print("Error: You can't divide by zero!")

    except ValueError:
        print("Error: Please enter valid numbers!")

    except NameError:
        print("Error: A variable is not defined!")

    else:
        print(f"The result of the division is: {result}")

    finally:
        print("Thank you for using the division program!")
divide_numbers()
```

Output:

```
Enter the numerator: 5
Enter the denominator: 0
ERROR!
Error: You can't divide by zero!
Thank you for using the division program!

=== Code Execution Successful ===

Enter the numerator: 5
```

```
Enter the denominator:
ERROR!
Error: Please enter valid numbers!
Thank you for using the division program!

=== Code Execution Successful ===
```

----------**----------**

Experiment 25: Write a program that opens a file and handles a FileNotFoundError exception if the file does not exist.

Program:

```
def read_file():
    try:
        file_name = input("Enter the name of the file you want to
open: ")

        with open(file_name, 'r') as file:
            content = file.read()
            print("\nFile content:")
            print(content)

    except FileNotFoundError:
        print(f"Error: The file '{file_name}' was not found. Please
check the file name and try again.")

    finally:
```

```
    print("\nThank you for using the file reader program!")

read_file()
```

Output:

```
Enter the name of the file you want to open: dfg
ERROR!
Error: The file 'dfg' was not found. Please check the file name and try again.

Thank you for using the file reader program!
```

----------****----------

Experiment 26: Write a program that executes an operation on a list and handles an IndexError exception if the index is out of range.

Program:

```
my_list = input("Enter a list of integers seprated by spaces: ")

try:
    my_list[len(my_list)]
except IndexError as e:
    print(f"Index error: {e}")
else:
    result = mylist[0]
    print(result)
```

Output:

Enter a list of integers seprated by spaces: 2 5 6 1
ERROR!
Index error: string index out of range

----------****----------

Experiment 27: Write a program that performs division and handles an `ArithmeticError` exception if there is a division-related error.

Program:

```
num1 = int(input("Enter first number: "))
num2 = int(input("Enter second number: "))

try:
    result = num1 // num2
except ArithmeticError as e:
    print(f"Arithmetic error: {e}")
else:
    print("Result:", result)
```

Output:

Enter first number: 6
Enter second number: 0ERROR!

Arithmetic Error: integer division or modulo by zero

----------****----------

Experiment 28: Write a program to print the reverse number pattern using a for loop.

Program:

```
num = int(input("Enter a positive integer: "))

for i in range(0, num + 1):
    for j in range(num - i, 0, -1):
        print(j, end=' ')
    print()
```

Output:

```
Enter a positive integer: 6
6 5 4 3 2 1
5 4 3 2 1
4 3 2 1
3 2 1
2 1
1
```

----------****----------

Experiment 29: Write a program to print all prime numbers within a given range

Program:

```
def is_prime(num):
```

```
    if num <= 1:
        return False
    for i in range(2, int(num**0.5) + 1):
        if num % i == 0:
            return False
    return True

def print_primes_in_range(start, end):
    print(f"Prime numbers between {start} and {end} are:")
    for num in range(start, end + 1):
        if is_prime(num):
            print(num, end=", ")
    print()

start = int(input("Enter the starting number of the range: "))
end = int(input("Enter the ending number of the range: "))

print_primes_in_range(start, end)
```

Output:

Enter the starting number of the range: 1
Enter the ending number of the range: 50
Prime numbers between 1 and 50 are:
2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47,

----------****----------

Experiment 30: Write a program to reverse the digits of a number.

Program:

```
def reverse_number(num):
    result = 0
    sign = 1 if num > 0 else -1

    while num > 0:
        digit = num % 10
        result = result * 10 + digit
        num = num // 10

    return sign * result

number = int(input("Enter a number: "))
print(f"Reversed number is : {reverse_number(number)}")
```

Output:

```
Enter a number: 56541651231
Reversed number is:  13215614565
```