`Problem 1` : Write a program that will give you in hand monthly salary after deduction on CTC - HRA(10%), DA(5%), PF(3%) and taxes deduction as below:

> Salary(Lakhs) : Tax(%)

- Below 5 : 0%
- 5-10 : 10%
- 10-20 : 20%
- aboove 20 : 30%

```
# Write code here
#write your code
ctc = int(input('Enter your anual CTC:'))

if ctc < 5000000:
    salary=ctc*.82
elif ctc<1000000:
    salary=ctc*.72
elif ctc<2000000:
    salary=ctc*.62
else:
    salary=ctc*.52

print("You in hand monthly salary will be-", round(salary/12,2))
```

```
Enter your anual CTC:1200000
You in hand monthly salary will be- 82000.0
```

`Problem 2` : Write a program that take a user input of three angles and will find out whether it can form a triangle or not.

Hint - Sum of all angles is 180 and all angles are positive

```
# Write code here
first = int(input('enter the 1st angle'))
second = int(input('enter the 2nd angle'))
third = int(input('enter the 3rd angle'))

if (first+second+third) == 180 and first>0 and second>0 and third>0:
  print('forms a triangle')
else:
  print('does not form a triangle')
```

```
enter the 1st angle30
enter the 2nd angle180
enter the 3rd angle-30
does not form a triangle
```

`Problem 3` : Write a program that will take user input of cost price and selling price and determines whether its a loss or a profit.

```
# Write code here
# write your code here
cost_price = int(input('Enter cost price-'))
selling_price = int(input('Enter selling price-'))

if cost_price < selling_price:
    print('Profit')
elif cost_price > selling_price:
    print('Loss')
else:
    print('No Loss No Gain')
```

```
Enter cost price-6000
Enter selling price-12000
Profit
```

`Problem 4` : Write a menu-driven program -

1. cm to ft

2. km to miles
3. USD to INR
4. exit

Hint

- 1 cm = 0.032ft
- 1km = 0.62
- 1 USD = 80 INR

```
# Write code here
menu = input("""
Hi select an option
1. cms to ft
2. km to miles
3. USD to INR
4. Exit
""")

if menu == '1':
  cm = float(input('enter the cm value'))
  print('ft value is',0.032*cm)
elif menu == '2':
  km = float(input('enter the km value'))
  print('miles value is',km*0.62)
elif menu == '3':
  usd = float(input('enter usd'))
  print('inr',usd*80)
else:
  exit()
```

```
Hi select an option
1. cms to ft
2. km to miles
3. USD to INR
4. Exit
4
```

`Problem 5` - Exercise 12: Display Fibonacci series up to 10 terms.

Note: The Fibonacci Sequence is a series of numbers. The next number is found by adding up the two numbers before it. The first two numbers are 0 and 1. For example, 0, 1, 1, 2, 3, 5, 8, 13, 21. The next number in this series above is 13+21 = 34

```
# Write code here

num1,num2 = 0,1

for i in range(10):
  print(num1)

  next = num1 + num2

  num1 = num2
  num2 = next
```

```
0
1
1
2
3
5
8
13
21
34
```

`Problem 6` - Find the factorial of a given number.

Write a program to use the loop to find the factorial of a given number.

The factorial (symbol: `!`) means to multiply all whole numbers from the chosen number down to 1.

For example: calculate the factorial of 5

```
5! = 5 × 4 × 3 × 2 × 1 = 120
```

Output:

```
120
```

```
# Write code here
num = int(input('enter the number'))

fact = 1
for i in range(1,num+1):
  fact = fact*i

print(fact)
```

```
enter the number5
120
```

Problem 7 - Reverse a given integer number.

Example:

Input:

```
76542
```

Output:

```
24567
```

```
# Write code here
number = int(input('enter the number'))

rev = 0

while number>0:
  last = number%10
  rev = rev*10 + last
  number = number//10

print(rev)
```

```
enter the number123456
654321
```

Problem 8 : Take a user input as integer N. Find out the sum from 1 to N. If any number if divisible by 5, then skip that number. And if the sum is greater than 300, don't need to calculate the sum further more. Print the final result. And don't use for loop to solve this problem.

Example 1:

Input:

```
30
```

Output:

```
276
```

```
# Write code here
N = int(input('enter the number'))
sum = 0
i = 1

while i < N+1:
  if i % 5 == 0:
    i+=1
    continue

  sum += i

  if sum > 300:
    sum = sum - i
    break

  i+=1

print(sum)
```

```
 enter the number50
 276
```

`Problem 9`: Write a program that keeps on accepting a number from the user until the user enters Zero. Display the sum and average of all the numbers.

```
# Write code here
sum = 0
count = 0

while True:
  num = int(input('enter number'))
  if num == 0:
    break
  sum = sum + num
  count = count + 1

print('sum',sum)
print('avg',sum/count)
```

```
 enter number1
 enter number2
 enter number3
 enter number4
 enter number5
 enter number0
 sum 15
 avg 3.0
```

`Problem 9`: Write a program which will find all such numbers which are divisible by 7 but are not a multiple of 5, between 2000 and 3200 (both included). The numbers obtained should be printed in a comma-separated sequence on a single line.

```
# Write code here
L = []
for i in range(2000,3201):
  if i % 7 == 0 and i % 5 != 0:
    L.append(str(i))

print(",".join(L))
```

```
2002,2009,2016,2023,2037,2044,2051,2058,2072,2079,2086,2093,2107,2114,2121,2128,2142,2149,2156,2163,2177,2184,2191,2198,
```

`Problem 10`: Write a program, which will find all such numbers between 1000 and 3000 (both included) such that each digit of the number is an even number. The numbers obtained should be printed in a space-separated sequence on a single line.

```
# Write code here
L = []
for i in range(1000,3001):
  flag = True

  curr = i

  while curr > 0:
    last = curr%10
    if last % 2 != 0:
      flag = False
      break
    curr = curr//10

  if flag == True:
    L.append(str(i))

print(",".join(L))
```

```
2000,2002,2004,2006,2008,2020,2022,2024,2026,2028,2040,2042,2044,2046,2048,2060,2062,2064,2066,2068,2080,2082,2084,2086,
```

Problem 11: A robot moves in a plane starting from the original point (0,0). The robot can move toward UP, DOWN, LEFT and RIGHT with a given steps.

The trace of robot movement is shown as the following:

```
UP 5
DOWN 3
LEFT 3
RIGHT 2
!
```

> The numbers after the direction are steps.

> ! means robot stop there.

Please write a program to compute the distance from current position after a sequence of movement and original point.

If the distance is a float, then just print the nearest integer.

Example:

```
Input:

UP 5
DOWN 3
LEFT 3
RIGHT 2
!

Output:

2
```

```
# Write code here
pos = [0,0]
while True:
  s = input('Enter the robot path')
  if s == '!':
    break
  direction = s.split()[0]
  steps = int(s.split()[1])

  if direction == 'UP':
    pos[1] = pos[1] + steps
  elif direction == 'DOWN':
    pos[1] = pos[1] - steps
  elif direction == 'LEFT':
    pos[0] = pos[0] - steps
  elif direction == 'RIGHT':
    pos[0] = pos[0] + steps
  else:
    pass

print('new pos',pos)
print((pos[0]**2 + pos[1]**2)**0.5)
```

```
Enter the robot pathUP 10
Enter the robot pathDOWN 5
Enter the robot pathLEFT 13
Enter the robot path!
new pos [-13, 5]
13.92838827718412
```

Problem 12 :Write a program to print whether a given number is a prime number or not

```
# Write code here
num = int(input('enter the num'))

flag = True
for i in range(2,num):
  if num%i == 0:
    flag = False
    break

if flag == True:
  print('Prime')
else:
  print('Not Prime')
```

```
enter the num21
Not Prime
```

Problem 13 :Print all the Armstrong numbers in a given range.

Range will be provided by the user
Armstrong number is a number that is equal to the sum of cubes of its digits. For example 0, 1, 153, 370, 371 and 407 are the Armstrong numbers.

```
# Write code here
start = int(input("Enter the start of the range: "))
end = int(input("Enter the end of the range: "))

for num in range(start, end+1):
    temp = num
    sum = 0
    order = len(str(num))
    while temp > 0:
        digit = temp % 10
        sum += digit ** order
        temp //= 10
    if num == sum:
        print(num)
```

```
Enter the start of the range: 0
Enter the end of the range: 200
0
1
2
3
4
5
6
7
8
9
153
```

Problem 14 :Calculate the angle between the hour hand and minute hand.

Note: There can be two angles between hands; we need to print a minimum of two. Also, we need to print the floor of the final result angle. For example, if the final angle is 10.61, we need to print 10.

Input:
H = 9 , M = 0
Output:
90
Explanation:
The minimum angle between hour and minute hand when the time is 9 is 90 degress.

```
# Write code here

# Write code here
h = int(input('Enter hours hand-'))
m = int(input('Enter minute hand-'))

# validate the input
if (h < 0 or m < 0 or h > 12 or m > 60):
    print('Wrong input')

# Idea is to minute angle - hour agnle from clockwise from 12 hour point

# 1 minute in minute angle manke 6 degree. (60 minute -> 360 degree)
m_angle = m*6

# every hour point yeilds to 30degree-- 12 hours-360degree plus if minute hand moves hour hands move too
# Every minute after hour hand take 0.5 degree movement. clockwise
h_angle = h*30 + m*0.5

# Take abs difference b/w them
angle = abs(h_angle - m_angle)


if angle>180:
    print(360-angle)
else:
    print(angle)
```

```
Enter hours hand-12
Enter minute hand-00
0.0
```

Explaination: This code calculates the angle between the hour and minute hand on a clock. It takes user input for the hour and minute hand positions, validates the input to make sure it is within the range of a clock (0-12 for hours and 0-60 for minutes), then calculates the angles of the hour and minute hand based on their position on the clock. It then finds the difference between the two angles and takes the absolute value

of that difference. It then checks if the angle is greater than 180, if it is, it prints the difference of 360 and the angle, else it prints the angle. The final output is the angle between the hands on the clock.

`Problem 15`:Given two rectangles, find if the given two rectangles overlap or not. A rectangle is denoted by providing the x and y coordinates of two points: the left top corner and the right bottom corner of the rectangle. Two rectangles sharing a side are considered overlapping. (L1 and R1 are the extreme points of the first rectangle and L2 and R2 are the extreme points of the second rectangle).

Note: It may be assumed that the rectangles are parallel to the coordinate axis.