Assignment - 7

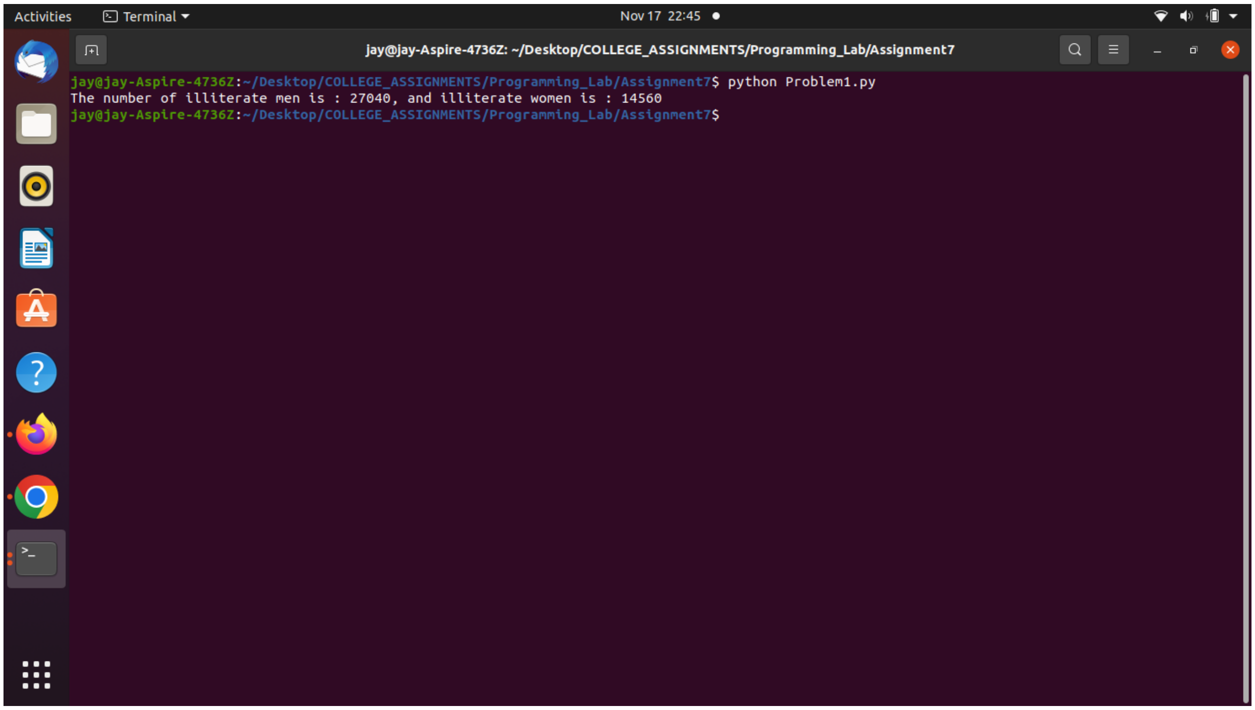
Name - Jay Shirgupe
PRN - 21510026

1. **In a town, the percentage of men is 52. The percentage of total literacy is 48. If total percentage of literate men is 35 of the total population, write a program in python to find the total number of illiterate men and women if the population of the town is 80,000.**

   Code -

```
total_population = 80000
percent_men = 52 / 100
literacy = 48 / 100
literacy_men = 35 / 100

percent_women = 1 - percent_men
number_men = percent_men * total_population
number_women = percent_women * total_population
total_literate = literacy * total_population
literate_men = literacy_men * number_men
literate_women = total_literate - literate_men
illiterate_men = number_men - literate_men
illiterate_women = number_women - literate_women

print("The number of illiterate men is : %d, and illiterate women is : %d"%
    (illiterate_men, illiterate_women))
```

   OUTPUT -



Activities
Terminal
Nov 17 22:45
jay@jay-Aspire-4736Z: ~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7

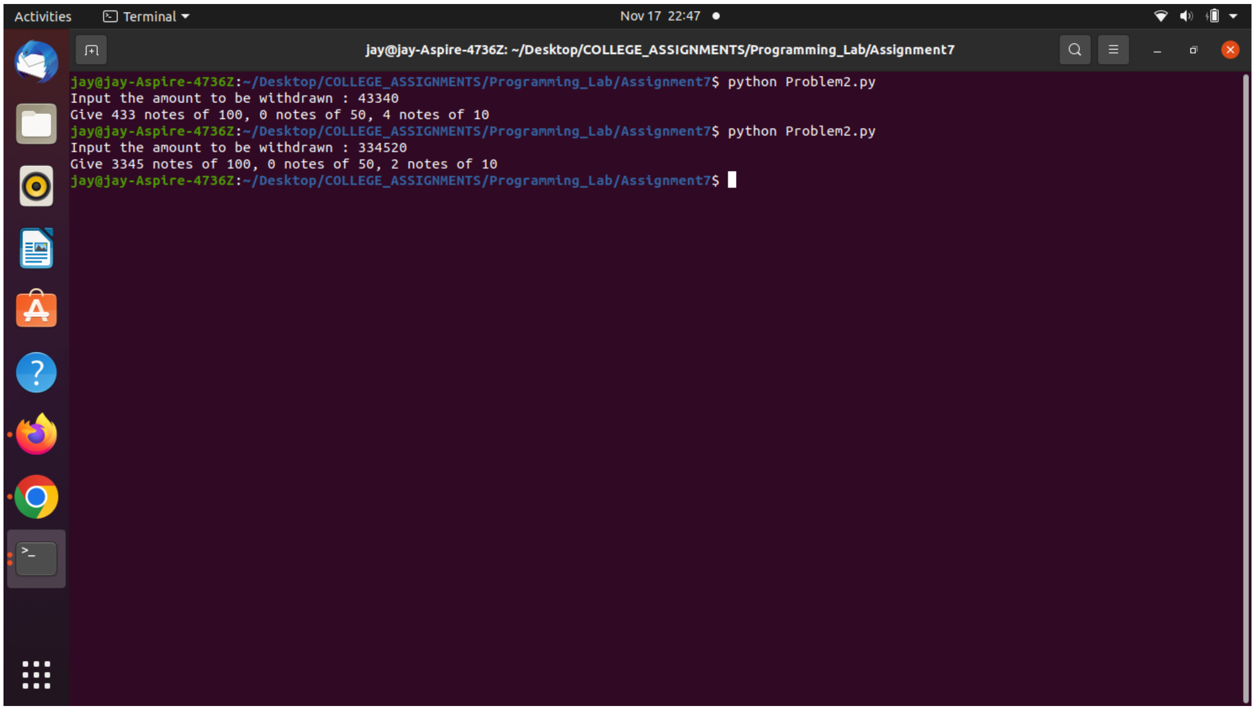
```
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$ python Problem1.py
The number of illiterate men is : 27040, and illiterate women is : 14560
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$
```

2. **A cashier has currency notes of denominations 10, 50 and 100. If the amount to be withdrawn is input through the keyboard in hundreds, find the total number of currency notes of each denomination the cashier will have to give to the withdrawer.**

Code -

```
def get_notes(n):
    hundred = n / 100
    n = n % 100
    fifty = n / 50
    n = n % 50
    ten = n / 10
    print("Give %d notes of 100, %d notes of 50, %d notes of 10" % (hundred, fifty, ten))

n = int(input("Input the amount to be withdrawn : "))
get_notes(n)
```

OUTPUT -



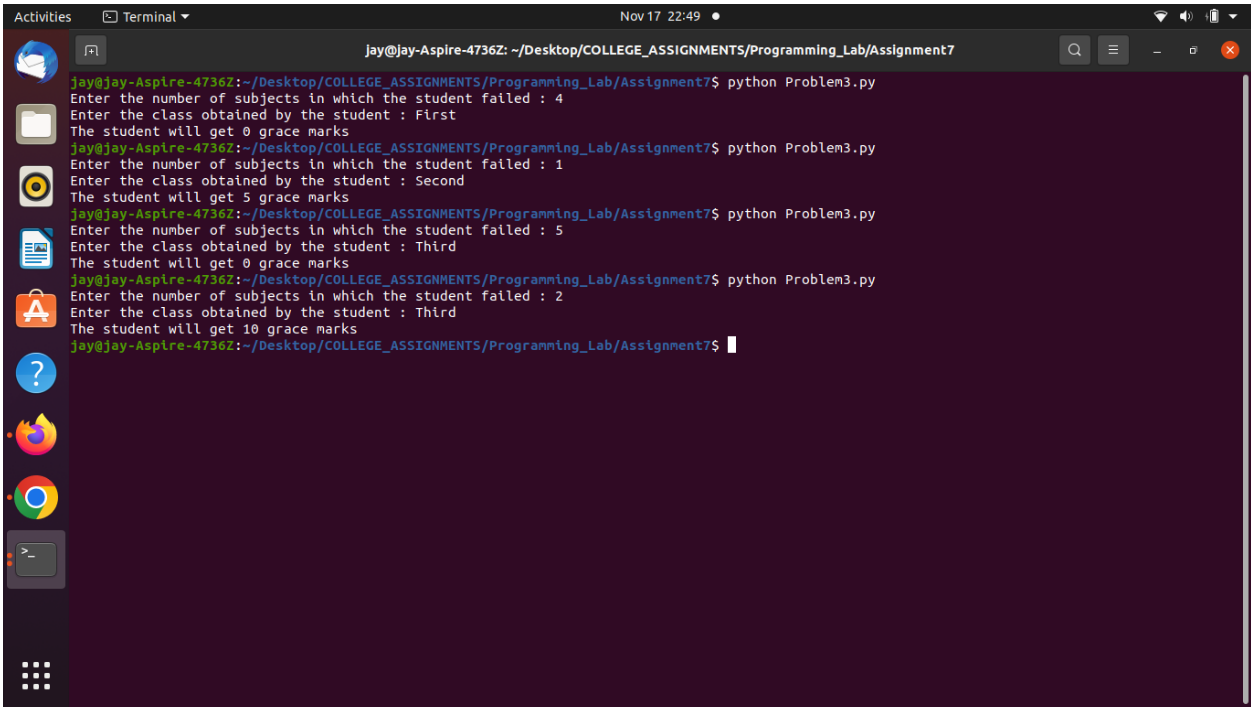
Activities
Terminal
Nov 17 22:47
jay@jay-Aspire-4736Z: ~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7

```
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$ python Problem2.py
Input the amount to be withdrawn : 43340
Give 433 notes of 100, 0 notes of 50, 4 notes of 10
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$ python Problem2.py
Input the amount to be withdrawn : 334520
Give 3345 notes of 100, 0 notes of 50, 2 notes of 10
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$
```

3. **Write a program which to find the grace marks for a student. The user should enter the class obtained by the student and the number of subjects he has failed in.**
   a) If the student gets first class and the number of subjects he failed in is greater than 3, then he does not get any grace.
   b) If the number of subjects he failed in is less than or equal to 3 then the grace is of 5 marks per subject.
   c) If the student gets second class and the number of subjects he failed in is greater than 2, then he does not get any grace.
   d) If the number of subjects he failed in is less than or equal to 2 then the grace is of 4 marks per subject.
   e) If the student gets third class and the number of subjects he failed in is greater than 1, then he does not get any grace.
   f) If the number of subjects he failed in is equal to 1 then the grace is of 5 marks per subject.
   Code -

```
def getGraceMarks(Class, numberFailSubjects):
    if Class == 'First' and numberFailSubjects > 3:
        return 0
    if numberFailSubjects <= 3:
        return (5 * numberFailSubjects)
    if Class == 'Second' and numberFailSubjects > 2:
        return 0
    if numberFailSubjects <= 2:
        return (4 * numberFailSubjects)
    if Class == 'Third' and numberFailSubjects > 1:
        return 0
    if numberFailSubjects == 1:
        return 5

n = int(input("Enter the number of subjects in which the student failed : "))
Class = input("Enter the class obtained by the student : ")
print("The student will get %d grace marks"%(getGraceMarks(Class, n)))
```

OUTPUT -


Activities
Terminal
Nov 17 22:49
jay@jay-Aspire-4736Z: ~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7

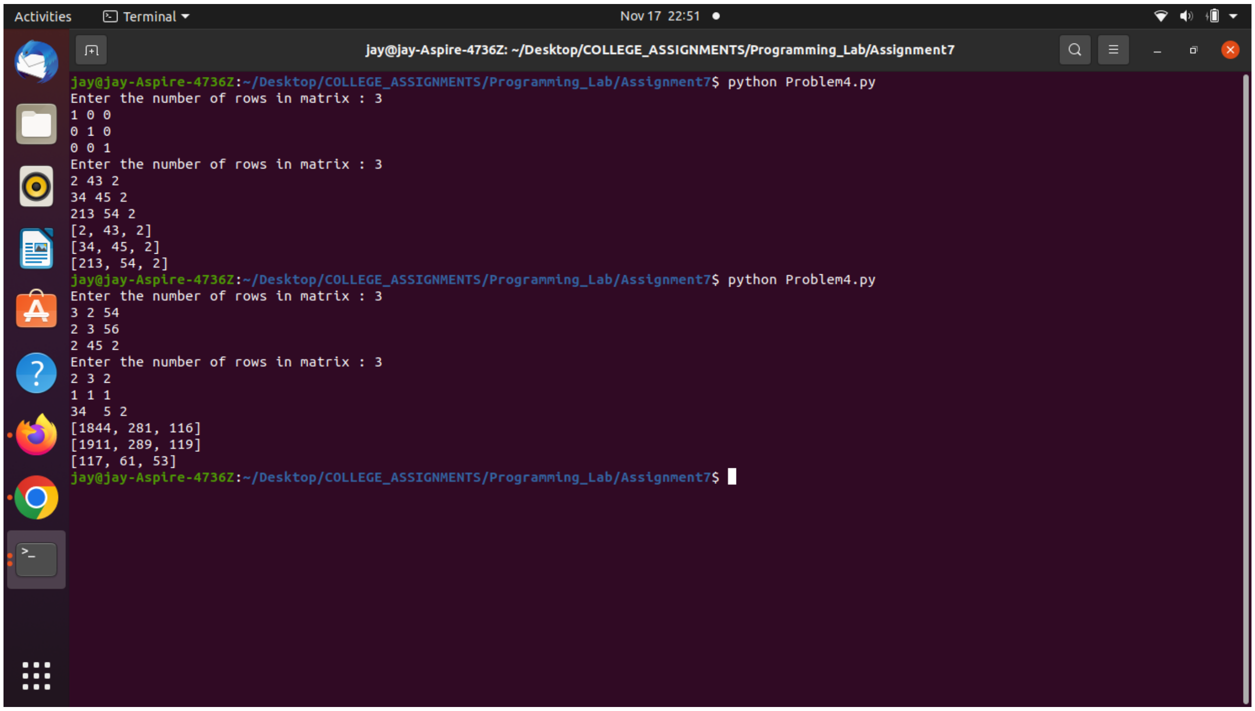
```
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$ python Problem3.py
Enter the number of subjects in which the student failed : 4
Enter the class obtained by the student : First
The student will get 0 grace marks
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$ python Problem3.py
Enter the number of subjects in which the student failed : 1
Enter the class obtained by the student : Second
The student will get 5 grace marks
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$ python Problem3.py
Enter the number of subjects in which the student failed : 5
Enter the class obtained by the student : Third
The student will get 0 grace marks
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$ python Problem3.py
Enter the number of subjects in which the student failed : 2
Enter the class obtained by the student : Third
The student will get 10 grace marks
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$
```

4. **Write a program in python for matrix multiplication.**

Code -

```
def multiply(A, B):
    if(len(A[0]) != len(B)):
        print('Matrix multiplicaton not possible')
        return False
    n = len(A)
    m = len(B[0])
    s = len(A[0])
    C = []
    for i in range(n):
        C.append([])
        for j in range(m):
            C[i].append(0)
            for k in range(s):
                C[i][j] += A[i][k] * B[k][j]
    return C

def printMatrix(M):
    for row in M:
        print(row)

def inputMatrix():
    M = []
    n = int(input("Enter the number of rows in matrix : "))
    for i in range(n):
        row = list(map(int, input().strip().split()))
        M.append(row)
    return M

A = inputMatrix()
B = inputMatrix()
printMatrix(multiply(A, B))
```

OUTPUT -

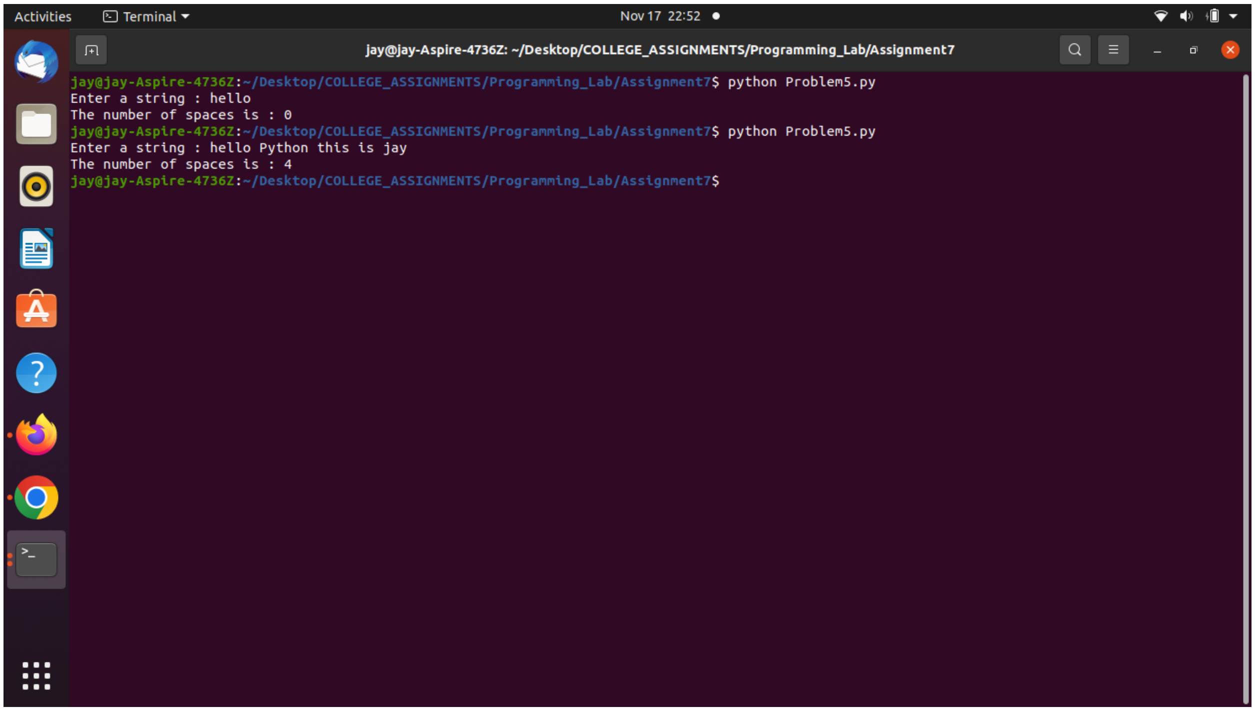

Activities
Terminal
Nov 17 22:51
jay@jay-Aspire-4736Z: ~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7

```
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$ python Problem4.py
Enter the number of rows in matrix : 3
1 0 0
0 1 0
0 0 1
Enter the number of rows in matrix : 3
2 43 2
34 45 2
213 54 2
[2, 43, 2]
[34, 45, 2]
[213, 54, 2]
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$ python Problem4.py
Enter the number of rows in matrix : 3
3 2 54
2 3 56
2 45 2
Enter the number of rows in matrix : 3
2 3 2
1 1 1
34  5 2
[1844, 281, 116]
[1911, 289, 119]
[117, 61, 53]
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$ 
```

5. **Write a loop that counts the number of space characters in a string. Recall that the space character is represented as ' '.**
Code -

```
def countSpaces(s):
    ans = 0
    for character in s:
        if character == ' ':
            ans += 1
    return ans

s = input("Enter a string : ")
print("The number of spaces is : %d"%countSpaces(s))
```

OUTPUT -

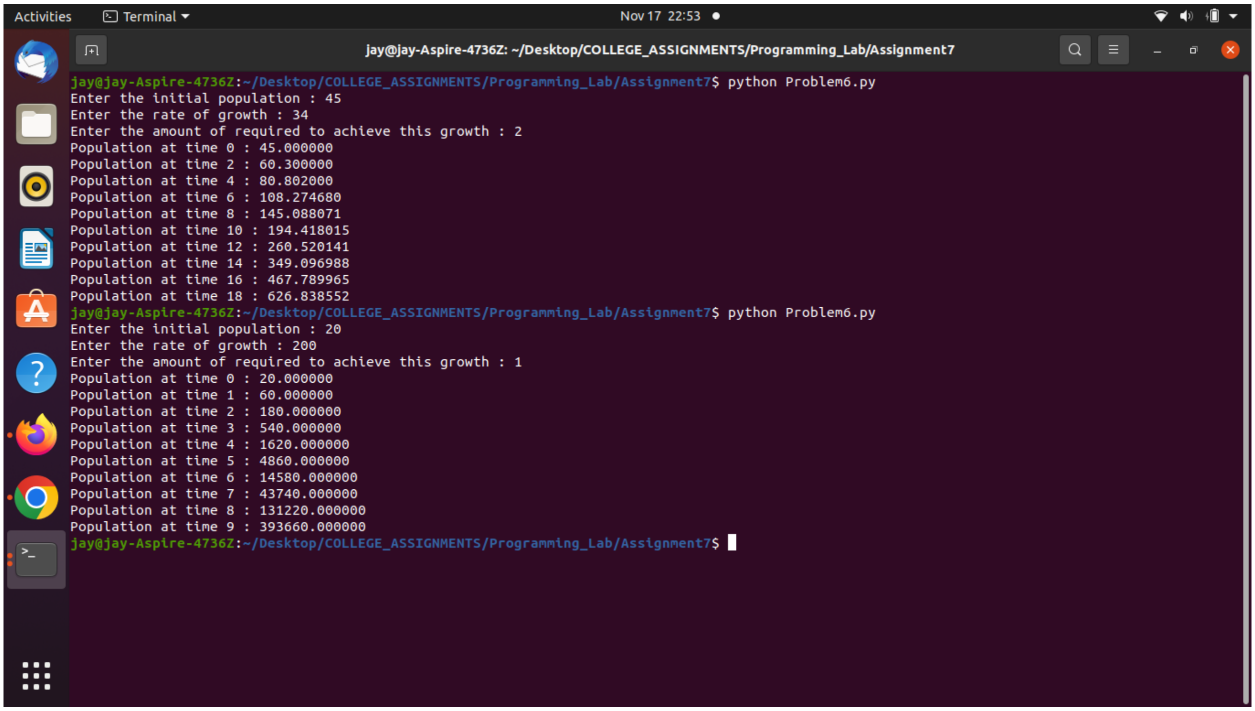

Activities
Terminal
Nov 17 22:52
jay@jay-Aspire-4736Z: ~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7

```
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$ python Problem5.py
Enter a string : hello
The number of spaces is : 0
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$ python Problem5.py
Enter a string : hello Python this is jay
The number of spaces is : 4
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$
```

6. **A local biologist needs a program to predict population growth. The inputs would be the initial number of organisms, the rate of growth (a real number greater than 0), the number of hours it takes to achieve this rate, and a number of hours during which the population grows. For example, one might start with a population of 500 organisms, a growth rate of 2, and a growth period to achieve this rate of 6 hours. Assuming that none of the organisms die, this would imply that this population would double in size every 6 hours. Thus, after allowing 6 hours for growth, we would have 1000 organisms, and after 12 hours, we would have 2000 organisms. Write a program that takes these inputs and displays a prediction of the total population.**
Code -

```
def getPopulation(initialPopulation, rateOfGrowth, growthPeriod, numberDoublings):
    ans = initialPopulation
    for i in range(numberDoublings):
        ans = ans * (1 + rateOfGrowth / 100)
    return ans

initialPopulation = int(input("Enter the initial population : "))
rateOfGrowth = int(input("Enter the rate of growth : "))
growthPeriod = int(input("Enter the amount of required to achieve this growth : "))

MAX = 10
for i in range(MAX):
    print("Population at time %d : %f" % (i * growthPeriod,
getPopulation(initialPopulation, rateOfGrowth, growthPeriod, i)))
```

OUTPUT -



jay@jay-Aspire-4736Z: ~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7

```
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$ python Problem6.py
Enter the initial population : 45
Enter the rate of growth : 34
Enter the amount of required to achieve this growth : 2
Population at time 0 : 45.000000
Population at time 2 : 60.300000
Population at time 4 : 80.802000
Population at time 6 : 108.274680
Population at time 8 : 145.088071
Population at time 10 : 194.418015
Population at time 12 : 260.520141
Population at time 14 : 349.096988
Population at time 16 : 467.789965
Population at time 18 : 626.838552
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$ python Problem6.py
Enter the initial population : 20
Enter the rate of growth : 200
Enter the amount of required to achieve this growth : 1
Population at time 0 : 20.000000
Population at time 1 : 60.000000
Population at time 2 : 180.000000
Population at time 3 : 540.000000
Population at time 4 : 1620.000000
Population at time 5 : 4860.000000
Population at time 6 : 14580.000000
Population at time 7 : 43740.000000
Population at time 8 : 131220.000000
Population at time 9 : 393660.000000
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$
```

7. **Write a python program to print following pattern as output.**

Code -

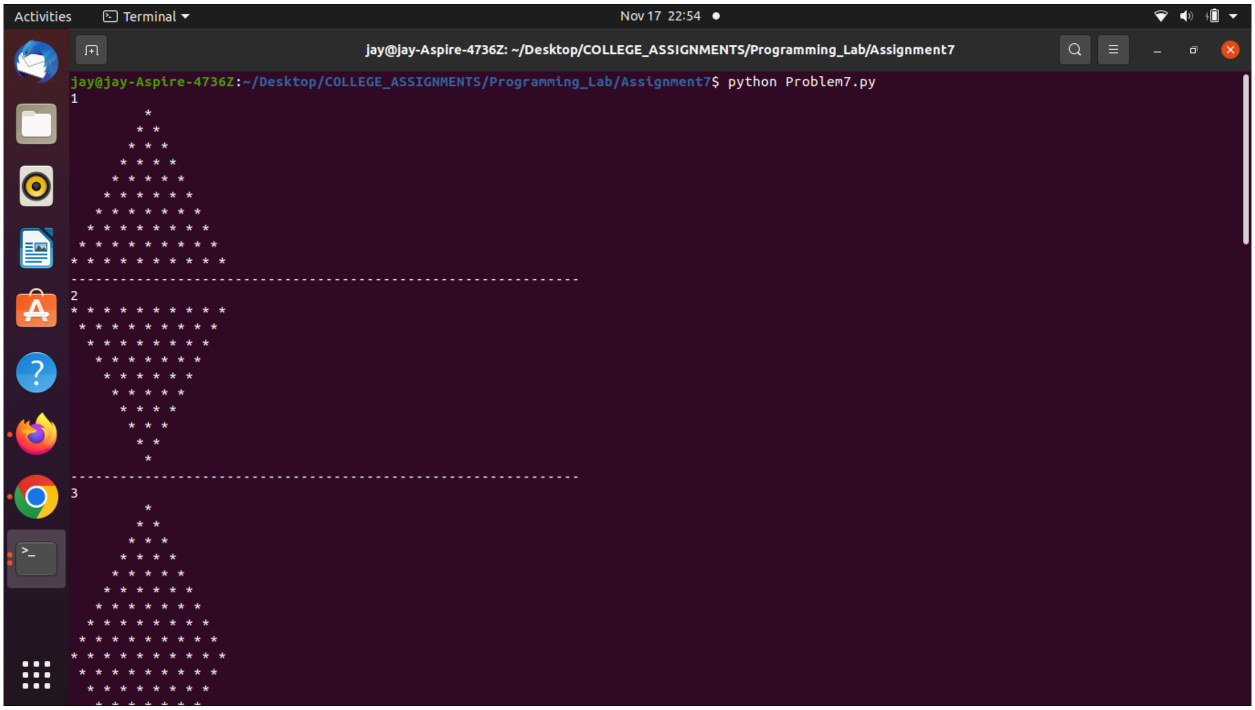
```
def print1(n):
    for i in range(1, n + 1):
        s = " " * (n - i)
        s += ("* " * i).strip()
        print(s)
```

```
def print2(n):
    for i in range(n, 0, -1):
        s = " " * (n - i)
        s += ("* " * i).strip()
        print(s)

def print3(n):
    print1(n)
    for i in range(n - 1, 0, -1):
        s = " " * (n - i)
        s += ("* " * i).strip()
        print(s)

def print4(n):
    m = 2 * n - 1
    arr = ['*'] * (2 * n - 1)
    cur1 = n - 1
    cur2 = n - 1
    for i in range(0, n):
        arr[cur1] = ' '
        arr[cur2] = ' '
        s = ' '.join(arr)
        print(s)
        cur1 = cur1 + 1
        cur2 = cur2 - 1
    cur1 = cur1 - 1
    cur2 = cur2 + 1
    for i in range(0, n - 1):
        arr[cur1] = '*'
        arr[cur2] = '*'
        s = ' '.join(arr)
        print(s)
        cur1 = cur1 - 1
        cur2 = cur2 + 1



def print5(n):
    for i in range(1, n + 1):
        s = "* " * i
        s.strip()
        print(s)
```

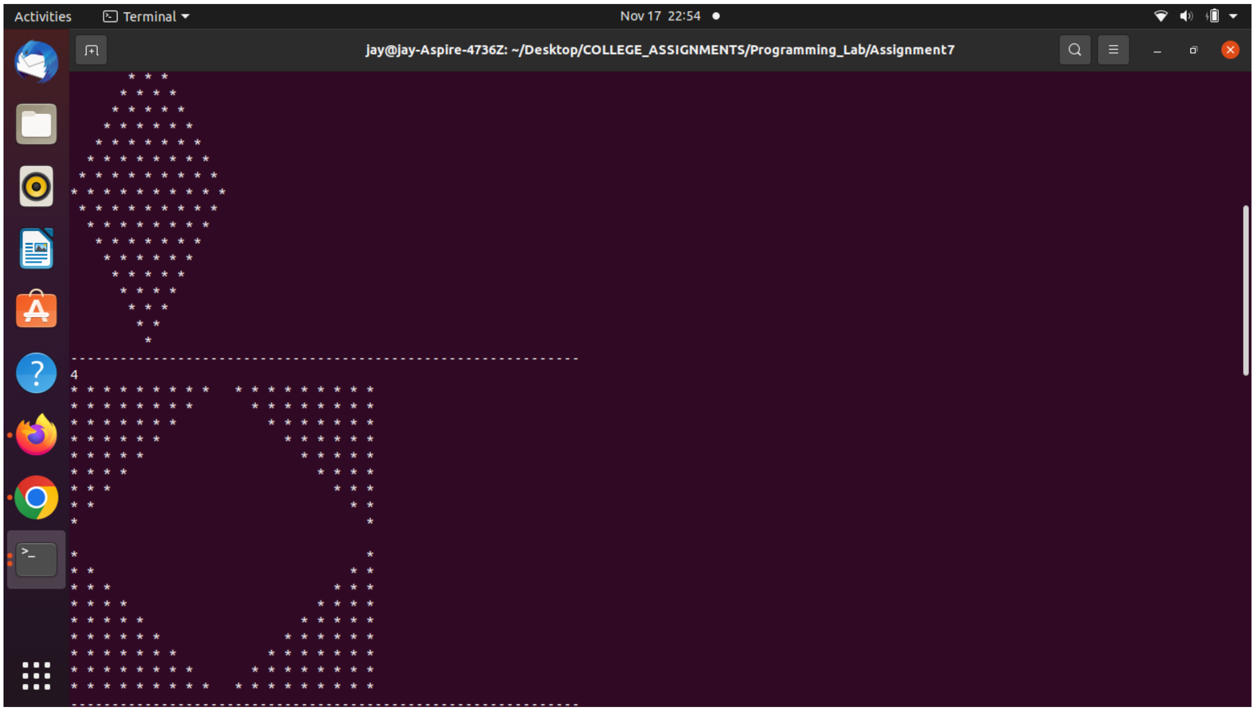
```
def print6(n):
    for i in range(n, 0, -1):
        s = "* " * i
        s.strip()
        print(s)

def print7(n):
    for i in range(1, n + 1):
        s = "  " * (n - i)
        s += ("* ") * i
        print(s)

def print8(n):
    for i in range(n, 0, -1):
        s = "  " * (n - i)
        s += ("* ") * i
        print(s)

def print9(n):
    m = 2 * n - 1
    arr = [' '] * (2 * n - 1)
    cur1 = n - 1
    cur2 = n - 1
    for i in range(0, n):
        arr[cur1] = '*'
        arr[cur2] = '*'
        s = ' '.join(arr)
        print(s)
        arr[cur1] = ' '
        arr[cur2] = ' '
        cur1 = cur1 + 1
        cur2 = cur2 - 1
    cur1 = cur1 - 2
    cur2 = cur2 + 2
    for i in range(0, n - 1):
        arr[cur1] = '*'
        arr[cur2] = '*'
        s = ' '.join(arr)
        print(s)
        arr[cur1] = ' '
        arr[cur2] = ' '
        cur1 = cur1 - 1
        cur2 = cur2 + 1
```

```
def printPat(n, b):
    if b == 1:
        print1(n)
    elif b == 2:
        print2(n)
    elif b == 3:
        print3(n)
    elif b == 4:
        print4(n)
    elif b == 5:
        print5(n)
    elif b == 6:
        print6(n)
    elif b == 7:
        print7(n)
    elif b == 8:
        print8(n)
    elif b == 9:
        print9(n)

n = 10
for i in range(1, 10):
    print(i)
    printPat(n, i)
    print("--------------------------------------------------------------")
```

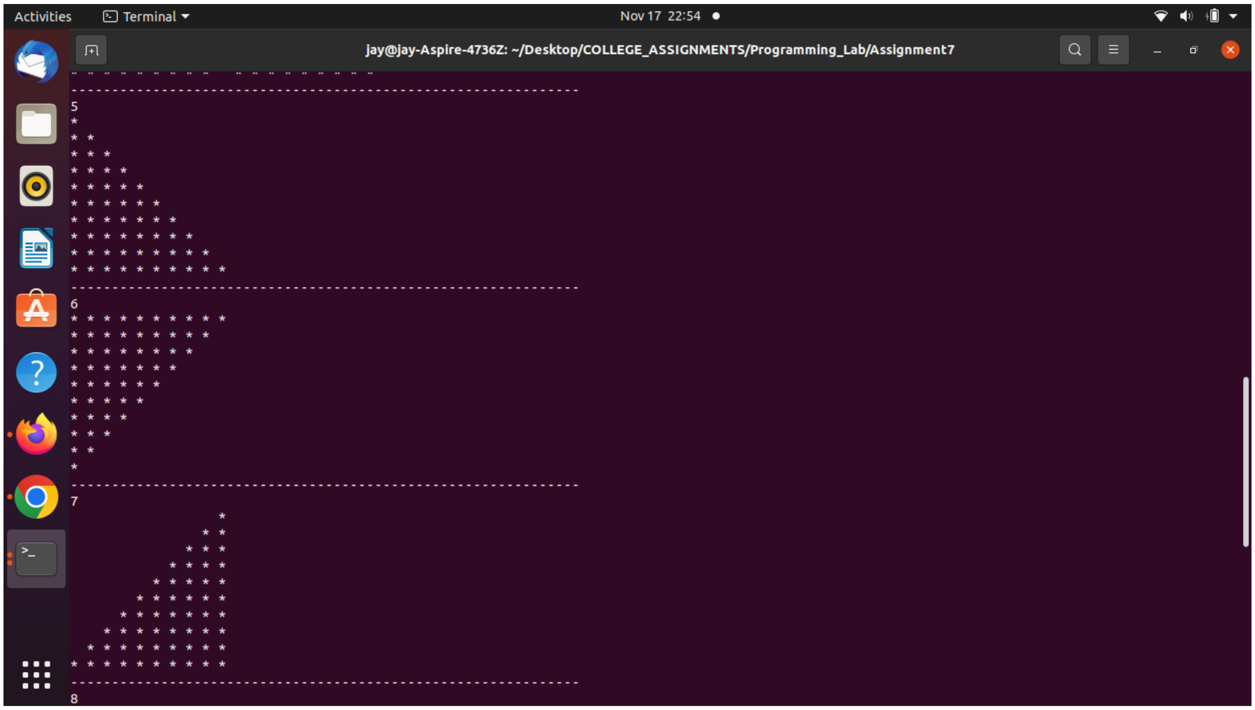
OUTPUT -


Activities
Terminal
Nov 17 22:54
jay@jay-Aspire-4736Z: ~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7

```
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$ python Problem7.py
1
         *
        * *
       * * *
      * * * *
     * * * * *
    * * * * * *
   * * * * * * *
  * * * * * * * *
 * * * * * * * * *
* * * * * * * * * *
------------------------------------------------------------
2
* * * * * * * * * *
 * * * * * * * * *
  * * * * * * * *
   * * * * * * *
    * * * * * *
     * * * * *
      * * * *
       * * *
        * *
         *
------------------------------------------------------------
3
         *
        * *
       * * *
      * * * *
     * * * * *
    * * * * * *
   * * * * * * *
  * * * * * * * *
 * * * * * * * * *
* * * * * * * * * *
 * * * * * * * * *
  * * * * * * * *
   * * * * * * *
```

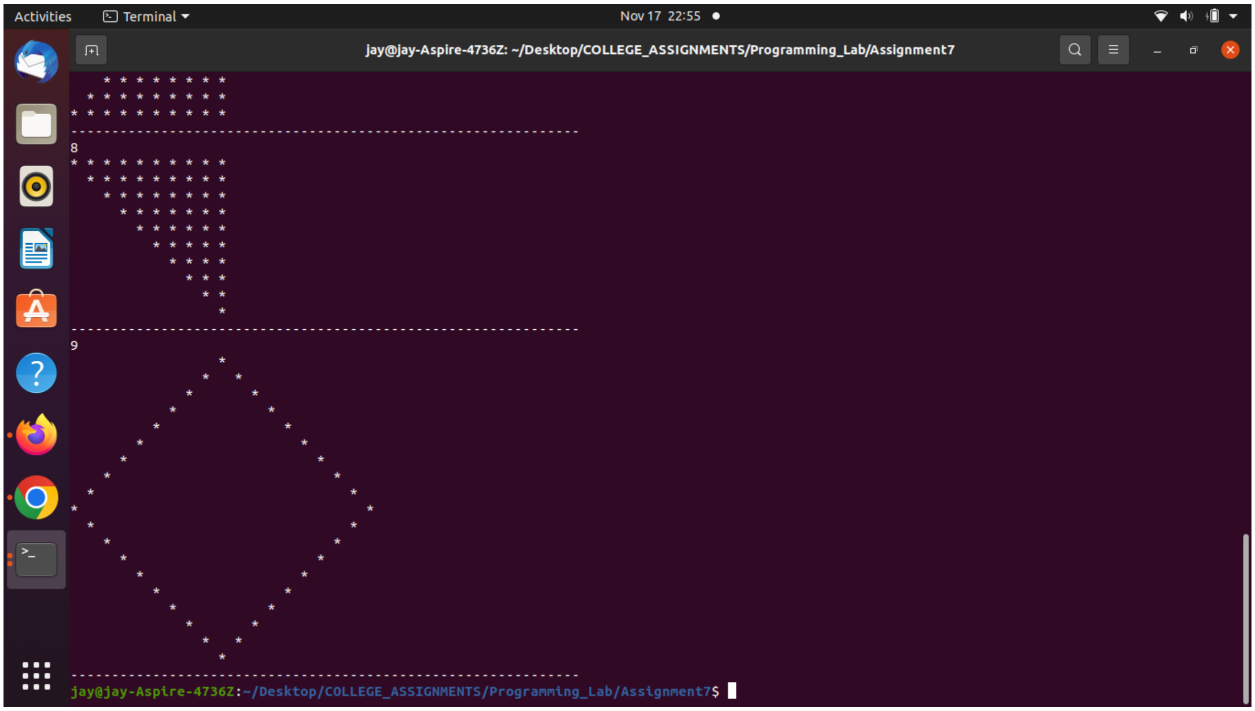

Activities
Terminal
Nov 17 22:54
jay@jay-Aspire-4736Z: ~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7

```
       * * *
      * * * *
     * * * * *
    * * * * * *
   * * * * * * *
  * * * * * * * *
 * * * * * * * * *
* * * * * * * * * *
 * * * * * * * * *
  * * * * * * * *
   * * * * * * *
    * * * * * *
     * * * * *
      * * * *
       * * *
        * *
         *
------------------------------------------------------------
4
* * * * * * * * *   * * * * * * * * *
* * * * * * * *       * * * * * * * *
* * * * * * *           * * * * * * *
* * * * * *               * * * * * *
* * * * *                   * * * * *
* * * *                       * * * *
* * *                           * * *
* *                               * *
*                                   *

*                                   *
* *                               * *
* * *                           * * *
* * * *                       * * * *
* * * * *                   * * * * *
* * * * * *               * * * * * *
* * * * * * *           * * * * * * *
* * * * * * * *       * * * * * * * *
* * * * * * * * *   * * * * * * * * *
------------------------------------------------------------
```



Activities
Terminal
Nov 17 22:54
jay@jay-Aspire-4736Z: ~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7

```
--------------------------------------------------------------
5
* 
* * 
* * * 
* * * * 
* * * * * 
* * * * * * 
* * * * * * * 
* * * * * * * * 
* * * * * * * * * 
* * * * * * * * * * 
--------------------------------------------------------------
6
* * * * * * * * * * 
* * * * * * * * * 
* * * * * * * * 
* * * * * * * 
* * * * * * 
* * * * * 
* * * * 
* * * 
* * 
* 
--------------------------------------------------------------
7
                  * 
                * * 
              * * * 
            * * * * 
          * * * * * 
        * * * * * * 
      * * * * * * * 
    * * * * * * * * 
  * * * * * * * * * 
* * * * * * * * * * 
--------------------------------------------------------------
8
```



Activities
Terminal
Nov 17 22:55
jay@jay-Aspire-4736Z: ~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7

```
    * * * * * * * * 
  * * * * * * * * * 
* * * * * * * * * * 
--------------------------------------------------------------
8
* * * * * * * * * * 
  * * * * * * * * * 
    * * * * * * * * 
      * * * * * * * 
        * * * * * * 
          * * * * * 
            * * * * 
              * * * 
                * * 
                  * 
--------------------------------------------------------------
9
                  * 
                *   * 
              *       * 
            *           * 
          *               * 
        *                   * 
      *                       * 
    *                           * 
  *                               * 
*                                   * 
  *                               * 
    *                           * 
      *                       * 
        *                   * 
          *               * 
            *           * 
              *       * 
                *   * 
                  * 
--------------------------------------------------------------
jay@jay-Aspire-4736Z:~/Desktop/COLLEGE_ASSIGNMENTS/Programming_Lab/Assignment7$ 
```