Міністерство освіти і науки України

Львівський національний університет ім. І.Я.Франка

**Звіт**

# Навчальна практика

# (python, C++)

Виконав:

Студент групи ПМп-11

Грищишин Роман

**Львів 2017**

Зміст

Завдання №1……………………………………………

Завдання №2……………………………………………

Завдання №3……………………………………………

Завдання №4……………………………………………

Завдання №5……………………………………………

Завдання №6……………………………………………

Завдання №7……………………………………………

Завдання №8……………………………………………

Завдання №9……………………………………………

Завдання №1

Умова:

### Для заданого n обчислити суму:

Короткий опис роботи програми:

### Виконуючи два цикли програма обраховує і сумує добутки.

Сирцевий код:

|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | | 1  2  3  4  5  6  7  8  9  10 | n = int(input('n= '))  summa = 0  multiplication = 1  for i in range(1, (n + 1)):  for k in range(1, (i\*2+1)):  multiplication \*= (k+1)  summa += multiplication  multiplication = 1    print("Сумма дорівнює", summa) | |

Приклад виконання:

### Input:

|  |
| --- |
| n= 10 |

Output:

|  |
| --- |
| Сумма дорівнює 51212944273488041646 |

Завдання №2

Умова:

### Використовуючи алгоритм Евкліда знайти найбільший спільний дільник двох чисел.

Короткий опис роботи програми:

### Програма рекурсивно шукає спільний дільник.

Сирцевий код:

|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | | 1  2  3  4  5  6  7  8  9  10  11  12  13 | def nsd(n, m):  if m == 0:  return n  elif n == 0:  return m  elif m > n:  return nsd(n, m % n)  else:  return nsd(n % m, m)  print(nsd(int(input("1'st num ")), int(input("2'nd num ")))) .    print("Сумма дорівнює", summa) | |

Приклад виконання:

### Input:

|  |
| --- |
| 1'st num 16  2'nd num 32 |

Output:

|  |
| --- |
| 16 |

Завдання №3

Умова:

### Прочитати дійсну матрицю nxn (n > 5), вияснити чи задана матриця є симетричною відносно головної діагоналі. Якщо ні – перевірити чи перестановкою рядків можна досягти заданої властивості.

Короткий опис роботи програми:

### Спершу, функція порівнює елементи на i j та j i позиціях, якщо вони відрізняються, то матриця не є симетричною. Далі, ми генеруємо всі перестановки рядків цієї матриці і виконуємо з ними ті самі дії.

Сирцевий код:

|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | | 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43 | import itertools  def output\_matrix(matrix):  for row in matrix:  for elem in row:  print(elem, end = " ")  else:  print()  def input\_matrix():  matrix = []  n = int(input("Введіть розмір матриці: "))  for m in range(n):  matrix.append([])  for row in range(len(matrix)):  for k in range(n):  print("Елемент {0}-го рядку, {1}-го стовпчика".format(row + 1, k + 1), end = " ")  matrix[row].append(int(input()))  return matrix  def perevirka(matrix):  for x in range(len(matrix)):  for m in range(len(matrix)):  if not(matrix[x][m] == matrix[m][x]):  return False  else:  return True  mat = input\_matrix()  print("Ваша матриця")  output\_matrix(mat)  if perevirka(mat):  print("Матриця симетрична відносно головної діагоналі")  else:  permutations = list(itertools.permutations(mat))  for mat in permutations:  if perevirka(mat):  print("Симетрії матриці відносно голоної діагоналі можна досягнути перестановкою рядків")  break  else:  print("Симетрії матриці відносно голоної діагоналі не можна досягнути перестановкою рядків") | |

Приклад виконання:

### Input:

|  |
| --- |
| Введіть розмір матриці: 3  Елемент 1-го рядку, 1-го стовпчика 3  Елемент 1-го рядку, 2-го стовпчика 3  Елемент 1-го рядку, 3-го стовпчика 3  Елемент 2-го рядку, 1-го стовпчика 2  Елемент 2-го рядку, 2-го стовпчика 3  Елемент 2-го рядку, 3-го стовпчика 3  Елемент 3-го рядку, 1-го стовпчика 2  Елемент 3-го рядку, 2-го стовпчика 3  Елемент 3-го рядку, 3-го стовпчика 2 |

Output:

|  |
| --- |
| Ваша матриця  3 3 3  2 3 3  2 3 2  Симетрії матриці відносно голоної діагоналі можна досягнути перестановкою рядків |

Завдання №4

Умова:

### Для заданог

Короткий опис роботи програми:

### Виконуючи два цикли програма обраховує і сумує добутки.

Сирцевий код:

|  |
| --- |
|  |

Приклад виконання:

### Input:

|  |
| --- |
|  |

Output:

|  |
| --- |
|  |

Завдання №1

Умова:

### Для заданог

Короткий опис роботи програми:

### Виконуючи два цикли програма обраховує і сумує добутки.

Сирцевий код:

|  |
| --- |
|  |

Приклад виконання:

### Input:

|  |
| --- |
|  |

Output:

|  |
| --- |
|  |