Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский государственный технический университет имени Н.Э. Баумана»

(национальный исследовательский университет)

Московский техникум космического приборостроения

**ЛАБОРАТОРНАЯ РАБОТА №2**

по теме: ОПТИМИЗАЦИЯ ПРОГРАММНОГО КОДА

Специальность 09.02.07 Информационные системы и программирование

Группа ТИП-61

|  |  |
| --- | --- |
| Проверил | Н.А. Сидорова |
| Разработал | С.С. Бобылёв |

Москва 2021

**СОДЕРЖАНИЕ**

1 Постановка задачи 3

2 Схема алгоритма программы 4

3 Листинг программы 6

4 Результаты выполнения программы 8

5 Ручной просчет 9

1. Постановка задачи
2. Получить элементы квадратной матрицы А={aij}(i=j=1,2,3,...,n), где n=6 , aij=2( j/2-3.1)(-2)-i (i-3.9).
3. Из матрицы А сформировать вектор X={Xi}(i=1,2,...,n) по правилу: Элементы вектора Х-среднее арифметическое элементов соответству-ющих строк матрицы А.
4. Используя вектор Х, получить значение переменной У по правилу:   
   Y=
5. Схема алгоритма программы





1. Листинг программы

/\*Программа laba2.py

Лабораторная работа №2 по профессиональному модулю МДК01.02 Поддержка и тестирование программных модулей

Тема "Оптимизация программного кода"

Язык: Python

Разработал Бобылев Сергей ТИП-61

Дата: 18.03.2021г.

ЗАДАНИЕ и ожидаемые входные данные:

Получить элементы квадратной матрицы А={aij}(i=j=1,2,3,...,n), где n=6 , aij=2( j/2-3.1)(-2)-i (i-3.9).

Из матрицы А сформировать вектор вектор X={Xi}(i=1,2,...,n) по правилу: Элементы вектора Х-среднее арифметическое элементов соответству-ющих строк матрицы А.

Используя вектор Х, получить значение переменной Y по правилу:

Y=(Xi+Xn-j+1)\*( Xi+1+Xn-(j+1)+1)\*…\* ( Xi+(n-1)+Xn-(j+(n-1))+1)

Переменные, используемые в программе:

a - массив;

x - вектор;

i - индекс строки;

j - индекс столбца;

per\_vich\_mas – переменная для вычисления массива;

Vivod\_mas - переменная для вывода матрицы;

Vivod\_vect - переменная для вывода вектора;

y - переменная, значение которой нужно найти;\*/

a,x=[],[]

Vivod\_mas,Vivod\_vect="",""

per\_vich\_mas,y=1,1

# Функция

# форматирования

# вывода

def tofixed(numobj,digits=4):

return f"{numobj:.{digits}f}"

# Вычисление

# матрицы и вектора

for i in range(1,7):

ai=[]

per\_vich\_mas\*=1/(-2)

per\_vich\_mas1=i-3.9

srednee\_arif=0

for j in range(1,7):

ai.append(float(tofixed(2\*(j/2-3.1)\*per\_vich\_mas\*per\_vich\_mas1)))

srednee\_arif+=ai[j-1]

a.append(ai)

x.append(float(tofixed(srednee\_arif/6)))

Vivod\_mas += f"{a[i - 1][0]:10} {a[i - 1][1]:10} {a[i - 1][2]:10} {a[i - 1][3]:10} {a[i - 1][4]:10} {a[i - 1][5]:10} \n"

Vivod\_vect += f"x[{i}] = {x[i - 1]}\n"

#Вычисление y

for i in range(1,7):

y\*=(x[i-1]+x[6-i])\*2

print("Вывод матрицы \n")

print(Vivod\_mas)

print("Вывод вектора \n")

print(Vivod\_vect)

print(f"y = {float(tofixed(y))}")

1. Результаты выполнения программы

Программа вычисляет элементы матрицы по формуле и записывает значения в двумерный массив, вывод матрицы изображен на рисунке 4.1.

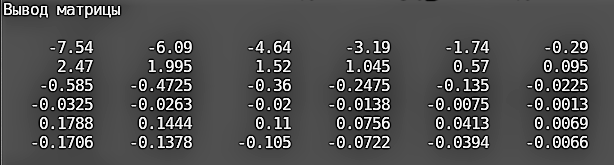


Рисунок 4.1 – Вывод матрицы

Затем программа вычисляет вектор Х с помощью матрицы, вывод результата на рисунке 4.2.

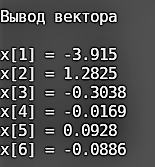


Рисунок 4.2 – Вывод вектора Х

В конце вычисляется значение Y по заданной формуле, полученный результат выводится на экран (рисунок 4.3).

y = 199.5603

Рисунок 4.3 – Вывод значения Y

1. Ручной просчет

Для проверки полученной матрицы использовался элемент А[23]. По данной формуле:

A[23]= 2(3/2-3.1)(-2)-2(2-3.9)= -3.2(-2)-2(-1.9)= -3.2×1/4×(-1.9)= -0.8×(-1.9)=1.52

Данное значение полностью совпадает со значением a[23]., представленном в результатах программы.

Для проверки полученного вектора использовался элемент X2. По данной формуле:

X2 = .

Данное значение полностью совпадает со значением Х2, представленном в результатах программы.

Для проверки полученного значения переменной Y необходимо выполнить просчет по формуле:

Данное значение полностью совпадает со значением Y, представленном в результатах программы.