**1. Задания для первой части практической работы   
МАТРИЧНЫЕ ОПЕРАЦИИ**

Для решения задачи используются типовые алгоритмы умножения векторов и матриц, которые студентам известны.

Операции умножения матриц и векторов не коммутативны, разичают умножение слева и умножение справа. Например, для двух матриц A и B результаты C1 = A\*B и C2 = B\*A за редкими исключениями отличаются. Принцип умножения векторов и матриц един – “строка на столбец”, т.е. каждый элемент результата равен сумме попарных произведений элементов строки первого объекта на соответсвующие элементы (если они имеются) столбца второго объекта.

Векторы рассматриваются как частные случаи матриц с одним столбцом или одной строкой, отсюда различают вектор-столбец и вектор-строку. В большинстве случаев эти различия не существенны, но иногда в операциях их умножения это необходимо учитывать. Схемы возможных операций, расчетные формулы получающихся результатов сведем в табл. 6.

Таблица 6

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Схема | операция | формула | примечание |
| 1 | \*  =  А  C  B | C = A \* B | Cij=∑k aik•bkj | результат матрица |
| 2 | \*  =  А  Y  X | Y = A \* X | Yi=∑j aij•Xj | результат  вектор-столбец |
| 3 | \*  =  X  Y  A | Y = X \* A | Yi=∑j Xj•Aij | результат  вектор-строка |
| 4 | •  =  X  C  Y | C = X \* Y | C = ∑ j Xj•Yj | результат скаляр |
| 5 | •  =  X  C  Y | C = Y \* X | Cij = Xj • Yi | редко, результат матрица |
| 6 | •  =  A  C  X | C = X \* A | Ci = Xj • Aii | редко, рез.  вектор-столбец |
| 7 | •  =  X  C  A | C = A \* X | Ci = Aii •Xj | редко, рез.  вектор-строка |

Для решения задачи 3 полезно иметь несколько разных процедур для сложения и умножения векторов и матриц. Для умножения матрицы A на вектор X будет полезна процедура **Mult(**int n,float \*\*a, float \*x,float \*y**),** где y – выходной вектор. Для умножения матриц могут быть полезны два разных варианта процедуры **Mult\_m1(**float \*\*a, float \*\*b, float \*\*c **),** где **c** – выходная матрица и **Mult\_m2(**float \*\*a, float \*\*b**)**. Последняя домножает матрицу **A** на **B**, при этом исходная матрица **A** ”портится” (изменяется).

Например, для решения матричной задачи **B=An**, возведения матрицы **А** в **n**-ю степень полезен следующий прием

copy\_matr(n,a,b); //копируем матрицу A в массив B

for(I=1; I<n; I++) mult\_m2(n,b,a); // домножаем матрицу B на A n-1 раз

// в результате получаем B=An, причем матрица A сохранилась

В других случаях более полезной может оказаться первый вариант попрограммы **Mult\_m1.**

### **Тестовое выражение для проверки правильности реализованного метода**

Постановка задачи: Вычислить **Z** = <операнд1> <операнд2> <операнд3>

где **Z** – результат операции (матрица, вектор или скалярная величина); операнды 1, 2, 3 представлены в табл. 3.

Правила конструирования формулы аналогичны. Например, для кода 013 варианта задача формулируется следующим образом:

|  |
| --- |
| Вычислить **Z** = **(А2-AТ) (АТ-А)2 D**  матрица \* матрица\*вектор получится вектор |

#### Таблица 3

|  |  |  |  |
| --- | --- | --- | --- |
| ЦВ | <операнд1> | <операнд2> | <операнд3> |
| 0 | **(А2-AТ)** | **E** | **Х**-столбец |
| 1 | **X** | **(АТ-А)2** | **X**-строка |
| 2 | **(А+Е)Т** | **D** | **D**-столбец |
| 3 | **(А+AТ)3** | **A** | **D**-строка |
| 4 | **P**-строка | **A(АТ-E)2** | **P**-столбец |
| 5 | **АТA2** | **1/D** | **P**-строка |
| 6 | **(АТA)2** | **АТ** | **SpA** |
| 7 | **1/X** | **Аn** | **SpA2** |
| 8 | **(АТ-E)2** | **TН** | **1/D**-строка |
| 9 | **A(АТ-E)** | **TВ** | **1/D**- столбец |

Где **E** - единичная матрица;

**X –** вектор полученный в задаче 2;

**D** - вектор из элементов главной диагонали матрицы А;

**P**- вектор, из элементов побочной диагонали матрицы А;

**1/X, 1/D** – векторы, составленные из обратных величин элементов **X** и **D**

**SpA**-след матрицы **А** (сумма элементов главной диагонали матрицы **А**);

**т** - верхний индекс для операции транспонирования матрицы;

**TН** – нижняя треугольная матрица: диагональные элементы =1, а над диагональю – 0:

**TВ** - верхняя треугольная матрица: аналогично, но под диагональю – 0

**TН** и **TВ** формируются из соответствующих элементов матрицы A