ЛАБОРАТОРНАЯ РАБОТА 12

Компьютерная математика
БГУ, ММФ, 1 курс, КМ
доц. Малевич А.Э.,
доц. Щеглова Н.Л.
ноябрь 2021

2 Компьютерные модели в задачах о точках и векторах

Задание 11. Постройте в *Mathematica* выражение, которое определяет, являются ли два вектора, заданные на плоскости, коллинеарными.

Выполнение задания 11

Рассмотрим два ненулевых вектора *a* и *b*, заданные своими координатами

$$a = \begin{pmatrix} a_x \\ a_y \end{pmatrix},\ b = \begin{pmatrix} b_x \\ b_y \end{pmatrix}.$$

Известно, что векторы *a* и *b* являются коллинеарными тогда и только тогда, когда существует $\lambda \in □$, $\lambda \neq 0$ такое, что $a = \lambda b$. Формально это условие можно представить в координатном виде $\frac{a_x}{b_x} = \frac{a_y}{b_y}$, откуда следует $a_x b_y - a_y b_x = 0$, или, используя понятие определителя матрицы

$$a_x b_y - a_y b_x = \begin{vmatrix} a_x & b_x \\ a_y & b_y \end{vmatrix} = 0.$$

Утверждение 1. Векторы *a* и *b* коллинеарны тогда и только тогда, когда определитель матрицы, содержащей координаты этих векторов, равен нулю, т. е.

$$\begin{vmatrix} a & b \end{vmatrix} = 0, \tag{3}$$

где *a* - столбец координат вектора *a* (a_x, a_y) и *b* - столбец координат вектора *b* (b_x, b_y).

Построим в *Mathematica* выражение вида (3). Рассмотрим два вектора, ассоциируем их с символами `V1` и `V2`

```
V1 = kmVector["a", {ax, ay}]
V2 = kmVector["b", {bx, by}]
kmVector[km, a, {ax, ay}]
kmVector[km, b, {bx, by}]
```

Тогда правую часть равенства (3) представим в *Mathematica*

```
Det[{V1["coord"], V2["coord"]}]
-ay bx + ax by
```
,

Булево выражение, возвращающее **True** в случае коллинеарности векторов **V1** и **V2**, и **False** в противном случае, запишем в виде

```
Det[{V1["coord"], V2["coord"]}] === 0
```

Постройте тест-функцию, которая для двух заданных векторов плоскости возвращает **True** в случае их коллинеарности и **False** в противном случае. Имя функции должно нести смысловую нагрузку. Напишите спецификацию функции. Тестируйте работу функции.

Задание 12. Постройте булеву функцию **isVectorsCollinear**, определяющую взаимную направленность двух заданных векторов плоскости. Функция возвращает одно из значений, соответствующих следующему расположению векторов:

1	коллинеарны и сонаправлены,
-1	коллинеарны и противонаправлены,
2	не коллинеарны.

Выполнение задания 12

Рассмотрим еще один способ определения коллинеарности векторов. Пусть на плоскости заданы два ненулевых вектора a и b. Выяснить, коллинеарны ли векторы a и b можно, используя определение операции скалярного произведения $a \circ b = |a||b|\cos(a,b)$ и тот факт, что косинус угла между коллинеарными векторами равен 1 или -1.

Утверждение 2. Для того чтобы векторы a и b были коллинеарными, необходимо и достаточно выполнения условия

$$|a \circ b| = |a||b|. \tag{4}$$

Утверждение 3. Векторы a и b со направлены при

$$a \circ b = |a||b|, \tag{5}$$

против направлены при

$$a \circ b = -|a||b|, \tag{6}$$

Для представления в *Mathematica* условий (4) – (6) подготовим выражение, результатом вычисления которого является список вида `{скалярное произведение заданных векторов, произведение длин заданных векторов}`. Полученный список позволит нам записывать любое из условий (4) – (6).

Рассмотрим два вектора

```
V1 = kmVector["a", {2, 4}]
V2 = kmVector["b", {3, 6}]
kmVector[km, a, {2, 4}]
kmVector[km, b, {3, 6}]
```

Для вычисления скалярного произведения запросим координаты этих векторов

```
#["coord"] & /@ {V1, V2}
{{2, 4}, {3, 6}}
```

и вычислим их скалярное произведение

```
Dot @@ (#["coord"] & /@ {V1, V2})
30
```

Аналогичное выражение позволяет вычислить произведение длин заданных векторов

```
Times @@ (#["length"] & /@ {V1, V2})
30
```

Нетрудно заметить, что выражения для вычисления скалярного произведения векторов и произведения длин векторов идентичны по своей структуре. Сведем их в одно выражение, оно должно возвращать результаты вычислений в списке вида

`{скалярное произведение векторов, произведение длин векторов}.`

Построим список, содержащий два списка: в первом будут содержаться координаты векторов, во втором – их длины

```
Outer[#2[#1] &, {"coord", "length"}, {V1, V2}]
```

$\{\{\{2, 4\}, \{3, 6\}\}, \{2\sqrt{5}, 3\sqrt{5}\}\}$

Тогда для вычисления скалярного произведения заданных векторов достаточно голову первого списка заменить на голову `Dot`

```
Dot @@
 Outer[#2[#1] &, {"coord", "length"}, {V1, V2}][[1]]
30
```

Чтобы вычислить произведение длин векторов, заменим голову второго списка, указав символ `Times`

```
Times @@
 Outer[#2[#1] &, {"coord", "length"}, {V1, V2}][[2]]
30
```

И снова очевидно, что последние два выражения можно объединить в одно

```
MapThread[
 Apply[#1,
   Outer[#2[#1] &, {"coord", "length"}, {V1, V2}][[
    #2]]] &, {{Dot, Times}, {1, 2}}]
{30, 30}
```

Мы получили список, первым элементом которого является скалярное произведение заданных векторов, вторым – произведение длин этих векторов. Теперь условие (4) коллинеарности двух заданных векторов можно записать в виде функции-однострочника

```
MapThread[                                                    (7)
  Apply[#1,
    Outer[#2[#1] &, {"coord", "length"}, {V1, V2}][[
     #2]]] &, {{Dot, Times}, {1, 2}}] //
 SameQ[Abs[First[#]], Last[#]] &
```

Выражение (7) является представлением выражения (4) в *Mathematica*. Эта модель может быть использована для построения теста на коллинеарность двух заданных векторов плоскости.

Постройте булеву функцию **isVectorsCollinear**, описанную в задании 12. Напишите спецификацию функции. Тестируйте работу функции.

Задание 13. Постройте булеву функцию **isVectorsOrthog**, определяющую, являются ли два заданных вектора плоскости перпендикулярными.

Выполнение задания 13

Задание 13 выполните самостоятельно, используя материал, представленный ниже.

Утверждение 4. Ненулевые векторы a и b взаимно перпендикулярны тогда и только тогда, когда выполняется условие

$$a \circ b = 0 \quad (8)$$

Следует помнить, что при представлении выражения (8) в *Mathematica* знак равенства соответствует встроенной булевой функции **SameQ**.

Задание 14. Постройте функцию **VectorsDirection**, определяющую взаимное направление двух векторов a и b плоскости. Функция возвращает одно из значений, соответствующих следующему взаимному направлению векторов a и b:

0 перпендикулярны,
1 коллинеарны и со направлены,
-1 коллинеарны и против направлены,
2 не коллинеарны и не перпендикулярны.

Выполнение задания 14

Выполнить самостоятельно.

Задание 15. Создайте функцию **VectorBisector**, вычисляющую единичный вектор, со направленный с биссектрисой угла между векторами a и b.

Выполнение задания 15

Направление биссектрисы угла между ненулевыми векторами a и b можно вычислить, если выполнить операцию сложения двух векторов, со направленных с заданными векторами a и b и имеющих одинаковую длину.

Уравнять длину и сохранить направление можно, умножая каждый из векторов на длину другого. После выполнения операции сложения векторов достаточно запросить орт результирующего вектора.

Для реализации модели в *Mathematica* введем два глобальных правила: правило сложения двух векторов и правило умножения вектора на число. Каждое из них ассоциируем с символом **kmVector**.

```
V1 = kmVector["a", {ax, ay}];
V2 = kmVector["b", {bx, by}];

VectsSum[V1_kmVector, V2_kmVector] ^:=
 kmVector[V1["coord"] + V2["coord"]]
VectsSum[V1, V2]
kmVector[km, , {ax + bx, ay + by}]

VectScale[V_kmVector, λ_] ^:= kmVector[λ V["coord"]]
VectScale[V1 , k]
kmVector[km, , {ax k, ay k}]
```

Теперь зададим векторы *a (2, 0)* и *b (0, 5)*, ассоциируя их с символами **V1** и **V2**

```
V1 = kmVector["a", {2, 0}];
V2 = kmVector["b", {0, 5}];
```

Получим векторы одинаковой длины, со направленные с заданными

```
VectScale[V2 , V1["length"]]
VectScale[V1 , V2["length"]]
kmVector[km, , {0, 10}]
kmVector[km, , {10, 0}]
```

Прежде чем выполнить операцию сложения двух векторов, запишем их списком

```
MapThread[VectScale[#1, #2["length"]] &,
 {{V1, V2}, {V2, V1}}]
{kmVector[km, , {10, 0}], kmVector[km, , {0, 10}]}
```

Тогда операция сложения векторов выполняется посредством функции **Apply**

```
VectsSum @@ MapThread[VectScale[#1, #2["length"]] &,
  {{V1, V2}, {V2, V1}}]
kmVector[km, , {10, 10}]
```

Таким образом, выражение, вычисляющее единичный вектор, со направленный с биссектрисой угла между векторами *a* и *b*, можно представить в виде

```
(VectsSum @@
   MapThread[VectScale[#1, #2["length"]] &,
    {{V1, V2}, {V2, V1}}])["ort"]
```

Постройте функцию **VectorBisector**, описанную в задании 15. Напишите спецификацию функции. Тестируйте работу функции.

Задание 16. Создайте функцию **VectorProjection**, вычисляющую вектор-проекцию вектора *a* на вектор *b*.

Выполнение задания 16

Векторной проекцией вектора *V* на вектор *W* называют вектор, коллинеарный вектору W и имеющий длину $\left\| V \right| \cos\alpha \right|$

$$pr_W V = \left| V \right| \cos\alpha W_e , \qquad (9)$$

где α - угол между векторами V и W, W_e - единичный вектор, сонаправленный с вектором W.

Вектор-проекцию вектора V на вектор W называют также **ортогональной составляющей вектора *V* вдоль вектора *W***.

Утверждение 5. Векторная проекция вектора V на вектор W может быть вычислена по формуле

$$pr_W V = (V, W_e) W_e \qquad (10)$$

где W_e - орт вектора W.

Покажем, что из выражения (9) следует (10). Используем определение скалярного произведения *(V, W)* векторов V и W, откуда получаем $\left| V \right| \cos\alpha = \dfrac{(V,W)}{\left| W \right|}$. С учетом того, что единичный вектор можно представить в виде $W_e = \dfrac{W}{\left| W \right|}$, получим $pr_W V = \dfrac{(V,W)}{\left| W \right|} W_e = (V, W_e) W_e$.

Представим в *Mathematica* выражение, правую часть равенства (10), которое вычисляет вектор-проекцию вектора V на вектор W. Для наглядности рассмотрим векторы

```
V = kmVector[{4, 4}];
W = kmVector[{0, 5}];
```

Получим координаты орта, единичного вектора, со направленного с вектором W

```
W@"ort"
kmVector[km, , {0, 1}]

(W@"ort")@"coord"
{0, 1}
```

Вычислим скалярное произведение векторов V и единичного вектора, со направленного с вектором W

```
Dot[V@"coord", #] &[(W@"ort")@"coord"]
```

Тогда выражение, вычисляющее вектор-проекцию вектора *V* на вектор *W*, можно представить в виде

```
VectScale[# , Dot[V@"coord",
     (#)@"coord"]] &[W@"ort"]
```

Постройте функцию **VectorProjection**, описанную в задании 16. Напишите спецификацию функции. Тестируйте работу функции.

Задание 17. Вектор *AB* повернут вокруг точки *A* на положительный угол φ, равный углу между векторами *AB* и *AC*. Определить координаты точки *C*, если известны координаты точек $A(x_A, y_A)$ и $B(x_B, y_B)$. [Бахвалов, с.50]

Выполнение задания 17

Математическое решение описано в [3, Бахвалов, с.50].

Полученное в [3] на с. 50 выражение (11) запишем, используя понятия матрицы и вектора

$$\begin{pmatrix} x_c \\ y_c \end{pmatrix} = \begin{pmatrix} x_a \\ y_a \end{pmatrix} + \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} x_b - x_a \\ x_b - x_a \end{pmatrix} \tag{11}$$

или, переходя к именам матриц и векторов,

$$pointC = pointA + Rotate(\varphi)\ vectorAB,$$

где *pointA*, *pointC* – вектор-столбцы координат точек *A*, *C* соответственно, *vectorAB* – вектор-столбец координат вектора *AB*, *Rotate*(φ) – матрица поворота вектора *AB* на положительный угол φ, равный углу между векторами *AB* и *AC*.

Построим в *Mathematica* выражение, вычисляющее координаты точки *C*, с учетом выражения (11) и введенной ранее базы знаний о точке и векторе.

Пусть на плоскости заданы две точки $A(x_A, y_A)$ и $B(x_B, y_B)$. Представим их в Mathematica, используя конструктор точки

```
{P1, P2} = MapThread[kmPoint[#1, #2] &,
  {{"A", "B"}, {{xA, yA}, {xB, yB}}}]
{kmPoint[km, A, {xA, yA}],
 kmPoint[km, B, {xB, yB}]}
```

Введем матрицу поворота вектора *AB* на положительный угол φ, равный углу между векторами *AB* и *AC*

```
VecRotate[ϕ_] :=
 {{Cos[ϕ], -Sin[ϕ]}, {Sin[ϕ], Cos[ϕ]}}
VecRotate[α] // MatrixForm
```

$$\begin{pmatrix} \text{Cos}[\alpha] & -\text{Sin}[\alpha] \\ \text{Sin}[\alpha] & \text{Cos}[\alpha] \end{pmatrix}$$

Получим координаты вектора, задавая его посредством конструктора

```
kmVector[P1, P2]@"coord"
{-xA + xB, -yA + yB}
```

Тогда выражение (12) представляет в *Mathematica* правую часть равенства (11) и позволяет вычислить искомые координаты точки *C*

```
P1@"coord" +                                        (12)
 VecRotate[ϕ].kmVector[P1, P2]@"coord"
{xA + (-xA + xB) Cos[ϕ] - (-yA + yB) Sin[ϕ],
 yA + (-yA + yB) Cos[ϕ] + (-xA + xB) Sin[ϕ]}
```

Создайте функцию пользователя, решающую задачу, сформулированную в задании 17. Дайте функции имя, несущее смысловую нагрузку. Напишите спецификацию построенной функции. Тестируйте работу функции.

Задание 18. Даны два вектора $a(x_a, y_a)$ и $b(x_b, y_b)$ с общей начальной точкой S. Определить угол φ, равный положительному углу между векторами *a* и *b* от вектора *a* до вектора *b*. [Бахвалов, с.51]

Выполнение задания 18

Математическое решение задачи описано в [3, Бахвалов, с.51]. Из рассуждений, приведенных в [3] на с. 51, следует, что косинус и синус угла φ, равный положительному углу между векторами *a* и *b*, от вектора *a* до вектора *b*, может быть вычислен из соотношений

$$\cos\phi = \frac{ax\,bx + ayby}{\sqrt{ax^2 + ay^2}\,\sqrt{bx^2 + by^2}} \tag{13}$$

$$\sin\phi = \frac{ax\,by - ay\,bx}{\sqrt{ax^2 + ay^2}\,\sqrt{bx^2 + by^2}}$$

Представим в *Mathematica* выражение (13). Зададим два экземпляра объекта «вектор на плоскости»

```
{V1, V2} = MapThread[kmVector[#1, #2] &,
  {{"a", "b"}, {{ax, ay}, {bx, by}}}]
{kmVector[km, a, {ax, ay}],
 kmVector[km, b, {bx, by}]}
```

Так же, как и при выполнении задания 12, построим список, первый элемент которого – список координат векторов, второй – список соответствующих длин векторов

```
Outer[#2[#1] &, {"coord", "length"}, {V1, V2}]
```

$$\left\{\{\{ax, ay\}, \{bx, by\}\}, \left\{\sqrt{ax^2 + ay^2}, \sqrt{bx^2 + by^2}\right\}\right\}$$

Тогда числитель выражения для вычисления косинуса искомого угла можно получить при помощи функции **Dot**, скалярного произведения векторов, а в знаменателе сформировать произведение длин

```
Outer[#2[#1] &, {"coord", "length"}, {V1, V2}] //
  (Apply[Dot, First[#]] / Times @@ Last[#]) &
```

$$\frac{ax\,bx + ay\,by}{\sqrt{ax^2 + ay^2}\,\sqrt{bx^2 + by^2}}$$

Для вычисления синуса искомого угла строим дробь, числитель которой – определитель матрицы координат векторов *a* и *b*, знаменатель – снова произведение длин

```
Outer[#2[#1] &, {"coord", "length"}, {V1, V2}] //
  (Det[First[#]] / Times @@ Last[#]) &
```

$$\frac{-ay\,bx + ax\,by}{\sqrt{ax^2 + ay^2}\,\sqrt{bx^2 + by^2}}$$

Теперь можно сформировать список, действуя последовательно каждой из построенных функций – получаем однострочник

```
#[Outer[#2[#1] &, {"coord", "length"},
    {V1, V2}]] & /@
  {(Apply[Dot, First[#]] / Times @@ Last[#]) &, (Det[First[#]] / Times @@ Last[#]) &}
```
(14)

$$\left\{\frac{ax\,bx + ay\,by}{\sqrt{ax^2 + ay^2}\,\sqrt{bx^2 + by^2}},\ \frac{-ay\,bx + ax\,by}{\sqrt{ax^2 + ay^2}\,\sqrt{bx^2 + by^2}}\right\}$$

Выражение (14) вычисляет косинус и синус угла φ, равного положительному углу между заданными векторами *a* и *b*, от вектора *a* до вектора *b*

Проводимые вычисления можно разбить на несколько выражений, чтобы увидеть промежуточные результаты. Для надежности вычислений оба выражения располагайте в одной ячейке

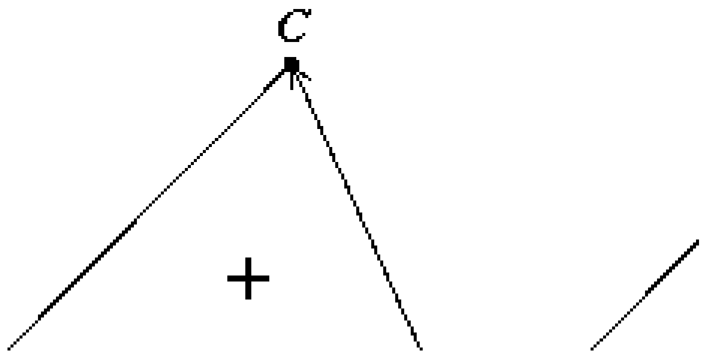
```
Outer[#2[#1] &, {"coord", "length"}, {V1, V2}]
#[%] & /@ {(Apply[Dot, First[#]] / Times @@ Last[#]) &,
   (Det[First[#]] / Times @@ Last[#]) &}
```

Создайте функцию пользователя, решающую задачу, сформулированную в задании 18. Дайте функции имя, несущее смысловую нагрузку. Напишите спецификацию построенной функции. Тестируйте работу функции.

Задание 19. Определить ориентированную площадь *S* треугольника *ABC*, если даны координаты его вершин $A(x_A, y_A)$, $B(x_B, y_B)$ и $C(x_C, y_C)$. Использовать понятие вектора. [Бахвалов, с.53]

Выполнение задания 19

Пусть треугольник, заданный координатами своих вершин, является ориентированным, т. е. задано также направление обхода вершин треугольника. Напомним, что многоугольник называют положительно ориентированным, если обход вершин треугольника совершается против часовой стрелки, и отрицательно ориентированным в случае обхода по часовой стрелке.



Рассмотрим стороны *AB* и *AC* треугольника как векторы, $AB = a(a_x, a_y)$ и $AC = b(b_x, b_y)$. Пусть φ – угол поворота вектора *a* до вектора *b* вокруг точки *A*. Тогда, по определению знака угла между векторами и ориентации треугольника, угол φ будет положительным в случае положительной ориентации треугольника и отрицательным в противном случае.

Известно, что площадь *S* треугольника *ABC* может быть вычислена по формуле

$$S = \frac{1}{2} |a|\,|b| \sin \phi$$

Подставляя значения длин векторов *a* и *b*, а также значение синуса угла из (13), получаем

$$S = \frac{1}{2} \sqrt{ax^2 + ay^2}\,\sqrt{bx^2 + by^2}\;\frac{ax\,by - ay\,bx}{\sqrt{ax^2 + ay^2}\,\sqrt{bx^2 + by^2}} =$$

$$= \frac{1}{2}(ax\,by - ay\,bx)$$

Тогда ориентированная площадь треугольника в направлении обхода *ABCA* вычисляется по формуле

$$S = \frac{1}{2}\begin{vmatrix} ax & ay \\ bx & by \end{vmatrix} = \frac{1}{2}\begin{vmatrix} x_A & y_A & 1 \\ x_B & y_B & 1 \\ x_C & y_C & 1 \end{vmatrix} \tag{15}$$

Если ориентация треугольника положительна, то $S > 0$, при отрицательной ориентации треугольника его площадь $S < 0$.

Представим в *Mathematica* выражение (15), вычисляющее ориентированную площадь треугольника.

Пусть заданы вершины треугольника $A(x_A, y_A)$, $B(x_B, y_B)$ и $C(x_C, y_C)$. Представим их как экземпляры объекта «точка на плоскости», ассоциируя с символами *P1*, *P2*, *P3* соответственно.

```
{P1, P2, P3} = MapThread[kmPoint[#1, #2] &,
  {{"A", "B", "C"}, {{xA, yA}, {xB, yB}, {xC, yC}}}]
{kmPoint[km, A, {xA, yA}],
 kmPoint[km, B, {xB, yB}], kmPoint[km, C, {xC, yC}]}
```

Построим матрицу, строки которой – координаты векторов $AB = a(a_x, a_y)$ и $AC = b(b_x, b_y)$.

```
kmVector[P1, #]@"coord" & /@ {P2, P3} // MatrixForm
```

$$\begin{pmatrix} -xA + xB & -yA + yB \\ -xA + xC & -yA + yC \end{pmatrix}$$

Тогда ориентированная площадь треугольника может быть получена посредством выражения

```
1/2 Det[kmVector[P1, #]@"coord" & /@ {P2, P3}]
```

$$\frac{1}{2}(-xB\,yA + xC\,yA + xA\,yB - xC\,yB - xA\,yC + xB\,yC)$$

Ориентированную площадь треугольника *ABC* можно также получить, введя проективные координаты точек *A*, *B*, *C*, записывая их в виде матрицы

```
Map[Append[#@"coord", 1] &, {P1, P2, P3}] // MatrixForm
```

$$\begin{pmatrix} xA & yA & 1 \\ xB & yB & 1 \\ xC & yC & 1 \end{pmatrix}$$

и вычисляя определитель этой матрицы

```
1/2 Det[Map[Append[#@"coord", 1] &, {P1, P2, P3}]]
```

$$\frac{1}{2}\left(-xB\ yA + xC\ yA + xA\ yB - xC\ yB - xA\ yC + xB\ yC\right)$$

Создайте функцию пользователя, решающую задачу, сформулированную в задании 19. Дайте функции имя, несущее смысловую нагрузку. Напишите спецификацию построенной функции. Тестируйте работу функции.

Литература

1. Голубева Л.Л., Малевич А.Э., Щеглова Н.Л. Компьютерная математика. Символьный пакет *Mathematica*. Лаб. практикум в 2 ч. Ч 1.- Минск: БГУ, 2012. – 235 с.
2. Никулин Е.А. Компьютерная геометрия и алгоритмы машинной графики.- СПб.: БХВ-Петербург, 2003. – 560 с.
3. Бахвалов С.В., Бабушкин Л.И., Иваницкая В.П. Аналитическая геометрия. – М., «Просвещение», 1970. – 376 с.