Лекция 1

Переменные

Скалярные

In[•]:=

Переменные рекомендуется называть строчными буквами, т.к. с прописных букв начинаются названия встроенных функций

In[•]:= `a = 1`

Out[•]= 1

По умолчанию Математика показывает (по возможности) точный результат выражений

In[•]:= `{1 / 153, √2}`

Out[•]= $\left\{\frac{1}{153}, \sqrt{2}\right\}$

Для вывода приближенного значения используется функция **N**

In[•]:=
```
N[1 / 153, 4]
N[1 / 153, 7]
```

Out[•]= 0.006536

Out[•]= 0.006535948

Если в выражении встречается вещественное число (с точкой), то и результат будет приближенный

In[•]:= `{ 1 / 153. , 1 / 153 }`

Out[•]= $\left\{0.00653595, \frac{1}{153}\right\}$

Переменные

Подавление вывода результата

```
In[•]:= a = 1;
```

```
In[•]:= {π, e, I, ∞, Degree, °}
Out[•]= {π, e, i, ∞, °, °}
```

```
In[•]:= {%, %%, %%%}
Out[•]= {{π, e, i, ∞, °, °}, 1, {0.00653595, 1/153}}
```

Palettes -> Basic Math Assistant

Очистка рабочей области

```
In[•]:= ClearAll[a]
        ClearAll["Global`*"]
        a =.
```

Ввод чисел с другим основанием

В двоичной системе

```
In[•]:= 2^^101
Out[•]= 5
```

В шестнадцатеричной системе

```
In[•]:= 16^^FF
Out[•]= 255
```

Функция BaseForm преобразовывает число из одной системы в другую

5 в десятичной системе преобразуется в двоичную

```
In[•]:= BaseForm[5, 2]
Out[•]//BaseForm=
101_2
```

222, записанное в двоичной системе преобразуется в десятичную систему

```
In[•]:= BaseForm[2^^11011110, 10]
Out[•]//BaseForm=
222
```

IntegerDigits

```
In[•]:= IntegerDigits[531]
Out[•]= {5, 3, 1}

In[•]:= N[1 / 153, 20]
Out[•]= 0.006535947712418300653 6
```

In[•]:= **RealDigits[1 / 153]**

Out[•]= {{{6, 5, 3, 5, 9, 4, 7, 7, 1, 2, 4, 1, 8, 3, 0, 0}}, −2}

Отложенное присвоение

```
In[•]:= a = c;
        a := c;
```

Выражение, записанное справа от знака отложенного присвоения вычисляется только тогда, когда это необходимо

```
In[•]:= a = c
Out[•]= c

In[•]:= c = 3;
        a
Out[•]= 3

In[•]:= a := 2 c + 10
        c = 1;
        a
Out[•]= 12

In[•]:= c = 2;
        a
Out[•]= 14
```

Еще один пример:

```
In[•]:= a := Random[]

In[•]:= a
Out[•]= 0.880576
```

Обращение к ***a*** приводит к вычислению нового случайного числа

```
In[•]:= a
Out[•]= 0.0424248
```

Двумерная запись

In[•]:= **b := 2**

c[Ctrl][-]3:=6

In[•]:= **c_3 := 6**

In[•]:= **a_3**

Out[•]= 0.872073_3

In[•]:= **a / 6**

Out[•]= 0.120457

a[Ctrl][/]6

In[•]:= $\frac{a}{6}$

Out[•]= 0.154271

[Ctrl][2]a

In[•]:= $\sqrt{a}$

Out[•]= 0.438446

Интервалы

In[•]:=
```
mass = Interval[{0.9, 1.1}];
work = Interval[{9, 10}];
velocity = Sqrt[2 work / mass]
```

Out[•]=
```
Interval[{4.0452, 4.71405}]
```

In[•]:=
```
velocity[[1]]
```

Out[•]=
```
{4.0452, 4.71405}
```

In[•]:=
```
Sqrt[2 * 9 / 1.1]
```

Out[•]=
```
4.0452
```

Списки

Одномерные

```
In[•]:= a = {10, 20, 30, 40, 50}
Out[•]= {10, 20, 30, 40, 50}

In[•]:= a[[2]]
Out[•]= 20

In[•]:= a[[{3, 4}]]
Out[•]= {30, 40}

In[•]:= Part[a, {3, 4}]
Out[•]= {30, 40}

In[•]:= a[[2 ;; 4]]
Out[•]= {20, 30, 40}
```

Списки

```
In[•]:= a = {10, 20, 30, 40, 50}
Out[•]= {10, 20, 30, 40, 50}

In[•]:= First[a]
Out[•]= 10

In[•]:= Last[a]
Out[•]= 50

In[•]:= Rest[a]
Out[•]= {20, 30, 40, 50}

In[•]:= Most[a]
Out[•]= {10, 20, 30, 40}

In[•]:= Drop[a, 2]
Out[•]= {30, 40, 50}

In[•]:= Select[a, 10 ≤ # ≤ 30 &]
Out[•]= {10, 20, 30}

In[•]:= f[x_] = 10 ≤ x ≤ 30;
        Select[a, f]
Out[•]= {10, 20, 30}

In[•]:=
```

Многомерные списки

```
In[•]:= a = {{10, 20, 30}, {40, 50, 60}, {80, 60, 30}, {41, 56, 67}}
Out[•]= {{10, 20, 30}, {40, 50, 60}, {80, 60, 30}, {41, 56, 67}}

In[•]:= MatrixForm[a]
Out[•]//MatrixForm=
(10  20  30
 40  50  60
 80  60  30
 41  56  67)

In[•]:= a[[2, 3]]
Out[•]= 60

In[•]:= MatrixForm[a[[{2, 3}, {1, 3}]]]
Out[•]//MatrixForm=
(40  60
 80  30)

In[•]:= MatrixForm[a[[{2, 3}, 1 ;; 3]]]
Out[•]//MatrixForm=
(40  50  60
 80  60  30)
```

Генерация списков

Range

In[•]:=
```
Range[5]
```

Out[•]=
```
{1, 2, 3, 4, 5}
```

In[•]:=
```
Range[3, 5]
```

Out[•]=
```
{3, 4, 5}
```

In[•]:=
```
Subdivide[0, 1, 5]
```

Out[•]= $\left\{0, \frac{1}{5}, \frac{2}{5}, \frac{3}{5}, \frac{4}{5}, 1\right\}$

In[•]:=

Генерация списков

Table

```
In[•]:= Table[i^2, {i, 2, 10}]
Out[•]= {4, 9, 16, 25, 36, 49, 64, 81, 100}

In[•]:= Table[i^2, {i, 2, 10, 2}]
Out[•]= {4, 16, 36, 64, 100}

In[•]:= Table[{i, i^2}, {i, 2, 10, 2}]
Out[•]= {{2, 4}, {4, 16}, {6, 36}, {8, 64}, {10, 100}}

In[•]:= Table[10 i + j, {i, 1, 3}, {j, 2, 8, 2}]
Out[•]= {{12, 14, 16, 18}, {22, 24, 26, 28}, {32, 34, 36, 38}}
```

Генерация списков

```
In[•]:= IdentityMatrix[3]
Out[•]= {{1, 0, 0}, {0, 1, 0}, {0, 0, 1}}

In[•]:= Table[If[i == j, 1, 0], {i, 1, 3}, {j, 1, 3}]
Out[•]= {{1, 0, 0}, {0, 1, 0}, {0, 0, 1}}

In[•]:= ConstantArray[3, 5]
Out[•]= {3, 3, 3, 3, 3}

In[•]:= Table[3, {5}]
Out[•]= {3, 3, 3, 3, 3}
```

Преобразование списков

```
In[•]:= a = Table[i^2, {i, 2, 10, 2}]
Out[•]= {4, 16, 36, 64, 100}

In[•]:= Reverse[a]
Out[•]= {100, 64, 36, 16, 4}

In[•]:= Join[{1, 2}, {3, 4}]
Out[•]= {1, 2, 3, 4}

In[•]:= {Min[a], Max[a]}
Out[•]= {4, 100}

In[•]:= {a + a, a * a, a / a}
Out[•]= {{8, 32, 72, 128, 200}, {16, 256, 1296, 4096, 10000}, {1, 1, 1, 1, 1}}

In[•]:= a = {3, 4};
        b = {1, 2};
        a^b
Out[•]= {3, 16}
```

Добавление и удаление элементов

```
In[•]:= a = Table[i^2, {i, 2, 10, 2}]
Out[•]= {4, 16, 36, 64, 100}

In[•]:= {Append[{1, 2, 3}, 30], Prepend[{1, 2, 3}, 30]}
Out[•]= {{1, 2, 3, 30}, {30, 1, 2, 3}}

In[•]:= {Insert[{1, 2, 3, 4}, 17, 2], Insert[{1, 2, 3, 4}, 17, -2]}
Out[•]= {{1, 17, 2, 3, 4}, {1, 2, 3, 17, 4}}

In[•]:= RotateLeft[a]
Out[•]= {16, 36, 64, 100, 4}

In[•]:= RotateRight[a]
Out[•]= {100, 4, 16, 36, 64}
```

Сортировка

```
In[•]:= a = RandomInteger[{-10, 10}, 4]
        Sort[a]
        f[a_, b_] := a > b;
        Sort[a, f]
        Sort[a, #1 > #2 &]
```

```
Out[•]= {10, 3, -6, 5}
```

```
Out[•]= {-6, 3, 5, 10}
```

```
Out[•]= {10, 5, 3, -6}
```

```
Out[•]= {10, 5, 3, -6}
```

```
In[•]:= data = {{x, 2}, {y, 1}, {z, 5}};
```

```
In[•]:= Sort[data, #1[[2]] < #2[[2]] &]
```

```
Out[•]= {{y, 1}, {x, 2}, {z, 5}}
```

```
In[•]:= a
        Ordering[a]
```

```
Out[•]= {10, 3, -6, 5}
```

```
Out[•]= {3, 2, 4, 1}
```

Матрицы

Изменение/замена строк и столбцов

In[•]:=
```
a = Table[0, {5}, {5}];
a // MatrixForm
```

Out[•]//MatrixForm=

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

In[•]:=
```
a[[All, 2]] = 1;
a[[ ;; , 2]] = 1;
a // MatrixForm
```

Out[•]//MatrixForm=

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

In[•]:=
```
a[[All, 2]] = Range[5];
a // MatrixForm
```

Out[•]//MatrixForm=

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 & 0 \end{pmatrix}$$

Матрицы

Блоки матриц

```
In[•]:= a = Table[0, {5}, {5}];
        a[[1 ;; 2, 3 ;; 4]] = {{1, 2}, {3, 4}};
        a // MatrixForm
```

Out[•]//MatrixForm=

$$\begin{pmatrix} 0 & 0 & 1 & 2 & 0 \\ 0 & 0 & 3 & 4 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

```
In[•]:= AppendTo[a, {7, 8, 9, 10, 11}]
        a // MatrixForm
```

```
Out[•]= {{0, 0, 1, 2, 0}, {0, 0, 3, 4, 0}, {0, 0, 0, 0, 0}, {0, 0, 0, 0, 0}, {0, 0, 0, 0, 0}, {7, 8, 9, 10, 11}}
```

Out[•]//MatrixForm=

$$\begin{pmatrix} 0 & 0 & 1 & 2 & 0 \\ 0 & 0 & 3 & 4 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 7 & 8 & 9 & 10 & 11 \end{pmatrix}$$

```
In[•]:= Do[AppendTo[a[[i]], 9], {i, 1, 6}]
        a // MatrixForm
```

Out[•]//MatrixForm=

$$\begin{pmatrix} 0 & 0 & 1 & 2 & 0 & 9 \\ 0 & 0 & 3 & 4 & 0 & 9 \\ 0 & 0 & 0 & 0 & 0 & 9 \\ 0 & 0 & 0 & 0 & 0 & 9 \\ 0 & 0 & 0 & 0 & 0 & 9 \\ 7 & 8 & 9 & 10 & 11 & 9 \end{pmatrix}$$

Partition & Tuples & Subsets

Partition

```
In[•]:= Partition[{a, b, c, d, e, f}, 2]
```

```
Out[•]= {{{{0, 0, 1, 2, 0, 9}, {0, 0, 3, 4, 0, 9}, {0, 0, 0, 0, 0, 9}, {0, 0, 0, 0, 0, 9}, {0, 0, 0, 0, 0, 9}, {7, 8, 9, 10, 11, 9}}, {1, 2}}, {2, d},
  {e, f}}
```

```
In[•]:= Partition[{a, b, c, d, e, f}, 2, 1]
```

```
Out[•]= {{{{0, 0, 1, 2, 0, 9}, {0, 0, 3, 4, 0, 9}, {0, 0, 0, 0, 0, 9}, {0, 0, 0, 0, 0, 9}, {0, 0, 0, 0, 0, 9}, {7, 8, 9, 10, 11, 9}}, {1, 2}},
  {{1, 2}, 2}, {2, d}, {d, e}, {e, f}}
```

Tuples

```
In[•]:= Tuples[{1, 2, 3}, 2]
```

```
Out[•]= {{1, 1}, {1, 2}, {1, 3}, {2, 1}, {2, 2}, {2, 3}, {3, 1}, {3, 2}, {3, 3}}
```

Subsets

```
In[•]:= Subsets[{1, 2, 3}, 2]
```

```
Out[•]= {{}, {1}, {2}, {3}, {1, 2}, {1, 3}, {2, 3}}
```

```
In[•]:= Subsets[{1, 2, 3}, {2}]
```

```
Out[•]= {{1, 2}, {1, 3}, {2, 3}}
```

```
In[•]:= Subsets[{{x1, y1}, {x2, y2}, {x3, y3}, {x4, y4}, {x5, y5}}, {3}]
```

```
Out[•]= {{{x1, y1}, {x2, y2}, {x3, y3}}, {{x1, y1}, {x2, y2}, {x4, y4}}, {{x1, y1}, {x2, y2}, {x5, y5}}, {{x1, y1}, {x3, y3}, {x4, y4}},
  {{x1, y1}, {x3, y3}, {x5, y5}}, {{x1, y1}, {x4, y4}, {x5, y5}}, {{x2, y2}, {x3, y3}, {x4, y4}}, {{x2, y2}, {x3, y3}, {x5, y5}},
  {{x2, y2}, {x4, y4}, {x5, y5}}, {{x3, y3}, {x4, y4}, {x5, y5}}}
```

Pattern

```
In[•]:= a = {1, 2, 3.0, 4, 5.0, 6, 7, {1, 3}};

In[•]:= Count[a, _List]

Out[•]= 1

In[•]:= Count[a, _Integer]

Out[•]= 5

In[•]:= Count[a, a_ /; a > 5 || a < 2]
        Cases[a, a_ /; a > 5 || a < 2]

Out[•]= 3

Out[•]= {1, 6, 7}

In[•]:= a = {"fd", 1, 2, 3, "4"}
        Position[a, _String]
        Cases[a, _String]

Out[•]= {fd, 1, 2, 3, 4}

Out[•]= {{1}, {5}}

Out[•]= {fd, 4}
```

Cases

Cases

```
In[•]:= Count[{y^3, 1, 2, 3, x^-1, x, x^2, x^3}, a_^b_ /; b == -1]
        Position[{y^3, 1, 2, 3, x^-1, x, x^2, x^3}, a_^b_ /; b == 2]
        Cases[{y^3, 1, 2, 3, x^-1, x, x^2, x^3}, a_^b_ /; b == 2]
```

```
Out[•]= 1
```

```
Out[•]= {{7}}
```

```
Out[•]= {x^2}
```

```
In[•]:= Table[Random[Real, {0, 100}, 3], {10}]
        Count[%, a_ /; a < 30]
```

```
Out[•]= {38.8, 40.4, 46.4, 16.4, 22.6, 80.3, 94.7, 87.5, 32.3, 26.6}
```

```
Out[•]= 3
```

Select

```
In[•]:= a = {1, 2, 3, 4, 5};
        Select[a, # > 3 &]
```

```
Out[•]= {4, 5}
```

Векторная алгебра

```
In[•]:= a = {1, 2, 3};
        b = {3, 4, 5};

In[•]:= a.b

Out[•]= 26

In[•]:= Cross[a, b]

Out[•]= {-2, 4, -2}

In[•]:= Norm[a]

Out[•]= √14

In[•]:= Projection[a, b]

Out[•]= {39/25, 52/25, 13/5}

In[•]:= (a.b)/Norm[b] (b/Norm[b])

Out[•]= {39/25, 52/25, 13/5}
```

Функции

Объявление функций

Встроенные функции Математики начинаются с большой буквы

```
In[•]:= {Sin[2], N[Sin[2]], Sin[2.0]}

Out[•]= {Sin[2], 0.909297, 0.909297}
```

```
In[•]:= hyp = Function[{x, y}, √(x^2 + y^2)]
Out[•]= Function[{x, y}, √(x^2 + y^2)]

In[•]:= hyp[1, 3]
Out[•]= √10

In[•]:= Clear[hyp]
        hyp[x_, y_] = √(x^2 + y^2)
Out[•]= √(x^2 + y^2)
```

x_ -- некое значение, которое мы будем называть при определении функции “x”

```
In[•]:= hyp[x_, y_] := √(x^2 + y^2)
```

Анонимная функция

Через анонимную функцию

In[•]:=
```
hyp = √(#1^2 + #2^2) &;
hyp[1, 2]
```

Out[•]= $\sqrt{5}$

Пример (почему это может быть удобно)

In[•]:=
```
√2 // N[#, 10] &
```

Out[•]= 1.414213562

Sort, Select

In[•]:=

In[•]:=

```
hyp = lambda x, y: math.sqrt(x**2+y**2)
```

Постфиксная, префиксная и инфиксная запись

Постфиксная запись

```
In[•]:= 12 // Sin
Out[•]= Sin[12]
```

```
In[•]:= √2 // N[#, 10] &
Out[•]= 1.414213562
```

Инфиксная запись

```
In[•]:= 2 ~ Plus ~ 1
Out[•]= 3
```

Префиксная запись

```
In[•]:= Sin@d
Out[•]= Sin[d]
```

Функциональная запись

```
In[•]:= Sin[1.2]
Out[•]= 0.932039
```

Функции с индексами

Функции с индексами

```
In[•]:= ClearAll[f]
        ClearAll[f]

In[•]:= f[1][x_, y_] = x + y;
        f[2][x_, y_] = x - y;

In[•]:= {f[1][1, 2], f[2][1, 2]}

Out[•]= {3, -1}

In[•]:= f1[x_, y_] = x + y;
        f2[x_, y_] = x - y;
        fRandomInteger[{1,2}][1, 2]

Out[•]= 3
```

Значения по умолчанию

In[•]:=
```
f[x_, y_ : 10] := 10 x + y;
f[10]
f[10, 20]
```

Out[•]=
```
True
```

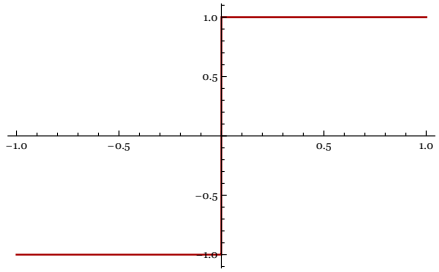
Out[•]=
```
False
```

Кусочные функции

In[•]:=
```
f[x_] := If[x > 0, 1, -1]
```

In[•]:=
```
g[x_] := 1 /; x > 0
g[x_] := -1 /; x ≤ 0
Plot[g[x], {x, -1, 1}]
```

Out[•]=



Замены

Замена

```
In[•]:= d + 3 + √(h + 1)
Out[•]= 3 + d + √(1 + h)

In[•]:= d + 3 + √(h + 1) /. {d → 3, h → 5}
Out[•]= 6 + √6
```

Показать в приближенном виде:

```
In[•]:= N[d + 3 + √(h + 1) /. {d → 3, h → 5}]
Out[•]= 8.44949

In[•]:= {a1, b1, c1, d1} /. a1 → 10
Out[•]= {10, b1, c1, d1}

In[•]:= {a1, b1, c1, d1} /. {a1 → 10, c1 → 20}
Out[•]= {10, b1, 20, d1}
```

Отложенная замена

```
In[•]:= {a1, a1, a1} /. a1 → RandomReal[]
        {a1, a1, a1} /. a1 :→ RandomReal[]
Out[•]= {0.54926, 0.54926, 0.54926}

Out[•]= {0.797451, 0.624844, 0.193939}

In[•]:=
```

Функции и списки

```
In[•]:= q = {1, 2, 3, 4, 5}
        Sin[q]
```

```
Out[•]= {1, 2, 3, 4, 5}
```

```
Out[•]= {Sin[1], Sin[2], Sin[3], Sin[4], Sin[5]}
```

```
In[•]:= f[x_] := x^2 + 3 * Sin[x]
        f[q]
```

```
Out[•]= {1 + 3 Sin[1], 4 + 3 Sin[2], 9 + 3 Sin[3], 16 + 3 Sin[4], 25 + 3 Sin[5]}
```

```
In[•]:= f =.
        f[x_] := Total[x]
        f[q]
```

```
Out[•]= 15
```

```
In[•]:= Map[f, {1, 2, 3}]
```

```
Out[•]= {1, 2, 3}
```

```
In[•]:= f /@ {1, 2, 3}
```

```
Out[•]= {1, 2, 3}
```

Apply

```
In[•]:= Apply[f, {1, 2, 3}]
        f @@ {1, 2, 3}
Out[•]= f[1, 2, 3]

Out[•]= f[1, 2, 3]
```

Apply для первого уровня

```
In[•]:= Apply[f, {{1, 2}, {2, 3}, {3, 4}}, {1}]
        f @@@ {{1, 2}, {2, 3}, {3, 4}}
Out[•]= {12, 23, 34}

Out[•]= {12, 23, 34}

In[•]:= Apply[ss[#2, #1] &, {{1, 2}, {2, 3}, {3, 4}}, {1}]
        ss[#2, #1] & @@@ {{1, 2}, {2, 3}, {3, 4}}
Out[•]= {ss[2, 1], ss[3, 2], ss[4, 3]}

Out[•]= {ss[2, 1], ss[3, 2], ss[4, 3]}

In[•]:= Total[{{1, 2}, {3, 4}, {5, 6, 7}}]
        Total: Lists of unequal length in {{1, 2}, {3, 4}, {5, 6, 7}} cannot be added.
Out[•]= Total[{{1, 2}, {3, 4}, {5, 6, 7}}]

In[•]:= Total[{{1, 2}, {3, 4}, {5, 6}}]
Out[•]= {9, 12}

In[•]:= #1^2 + #2 * 10 & @@@ {{1, 2}, {3, 4}, {5, 6}}
Out[•]= {21, 49, 85}
```

Map

```
In[•]:= Map[Apply[Plus, #] &, {{1, 2}, {3, 4}, {5, 6, 7}}]
Out[•]= {3, 7, 18}

In[•]:= Apply[Plus, #] & /@ {{1, 2}, {3, 4}, {5, 6, 7}}
Out[•]= {3, 7, 18}
```

MapThread

```
In[•]:= MapThread[fun, {{p1, p2, p3}, {1, 2, 3}}]
Out[•]= {fun[p1, 1], fun[p2, 2], fun[p3, 3]}

In[•]:= MapThread[fun, {{p1, p2, p3}, {1, 2, 3}, {10, 20, 30}}]
Out[•]= {fun[p1, 1, 10], fun[p2, 2, 20], fun[p3, 3, 30]}
```

Центр масс системы точек

```
In[•]:= r = {{1, 2}, {3, 2}, {4, 6}, {1, 6}};
        m = {1, 3, 2, 1};

In[•]:= Total[MapThread[#1 * #2 &, {r, m}]] / Total[m]

Out[•]= {19/7, 26/7}

In[•]:= Total[r]

Out[•]= {9, 16}

In[•]:= Total[r, 2]

Out[•]= 25

In[•]:= Total[r, {2}]

Out[•]= {3, 5, 10, 7}
```

Вариант 2

```
In[•]:= Transpose[{r, m}]

Out[•]= {{{1, 2}, 1}, {{3, 2}, 3}, {{4, 6}, 2}, {{1, 6}, 1}}

In[•]:= Total[Map[#[[1]] * #[[2]] &, Transpose[{r, m}]]] / Total[m]

Out[•]= {19/7, 26/7}
```

Вариант 3

```
In[•]:= Total[r * m] / Total[m]

Out[•]= {19/7, 26/7}
```

Код в процедурном стиле

In[•]:=
```
rc = 0;
For[i = 1, i ≤ Length[r], i++,
 rc = rc + (r[[i]] * m[[i]]) / Total[m]
]
rc
```

Out[•]= $\left\{\frac{19}{7}, \frac{26}{7}\right\}$

Расстояние между парами точек

Даны координаты точек на плоскости. Необходимо определить пару точек, расстоянием между которыми максимально.

```
In[•]:= r = {{1, 2}, {3, 2}, {4, 6}, {1, 6}};

        pairs = Subsets[r, {2}]
Out[•]= {{{1, 2}, {3, 2}}, {{1, 2}, {4, 6}}, {{1, 2}, {1, 6}}, {{3, 2}, {4, 6}}, {{3, 2}, {1, 6}}, {{4, 6}, {1, 6}}}

In[•]:= pairs[[1]]
Out[•]= {{1, 2}, {3, 2}}

In[•]:= Apply[Subtract, pairs[[1]]]
Out[•]= {-2, 0}

In[•]:= Apply[Subtract, pairs[[1]]]^2
Out[•]= {4, 0}

In[•]:= Sqrt[Total[Apply[Subtract, pairs[[1]]]^2]]
Out[•]= 2
```

Расстояние между парами точек

```
In[•]:= d = Map[Sqrt[Total[Apply[Subtract, #]^2]] &, pairs] // N
        Ordering[d]
Out[•]= {2., 5., 4., 4.12311, 4.47214, 3.}
Out[•]= {1, 6, 3, 4, 5, 2}
In[•]:= d = Map[Sqrt[Total[Apply[Subtract, #]^2]] &, pairs]
        Ordering[d]
Out[•]= {2, 5, 4, Sqrt[17], 2 Sqrt[5], 3}
Out[•]= {1, 6, 3, 2, 5, 4}
In[•]:= Max[d]
Out[•]= 5
In[•]:= pairs[[Ordering[d][[-1]]]]
Out[•]= {{3, 2}, {4, 6}}
```

In[•]:= **TreeForm[d]**

Out[•]//TreeForm=