

# Лекция 1

# Переменные

---

## Скалярные

In[322]:=

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

In[323]:= **a = 1**

Out[323]= 1

По умолчанию Математика показывает (по возможности) точный результат выражений

In[324]:=  $\{1/153, \sqrt{2}\}$

Out[324]=  $\left\{\frac{1}{153}, \sqrt{2}\right\}$

Для вывода приближенного значения используется функция **N**

In[325]:= **N[1/153, 4]**

**N[1/153, 7]**

Out[325]= 0.006536

Out[326]= 0.006535948

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

In[327]:=  $\{1/153., 1/153\}$

Out[327]=  $\left\{0.00653595, \frac{1}{153}\right\}$

# Переменные

---

## Подавление вывода результата

In[328]:= **a = 1;**

In[329]:= { $\pi$ , e, I,  $\infty$ , Degree,  $^\circ$ }

Out[329]= { $\pi$ , e, i,  $\infty$ ,  $^\circ$ ,  $^\circ$ }

In[330]:= {%, %, %%%}

Out[330]= { { $\pi$ , e, i,  $\infty$ ,  $^\circ$ ,  $^\circ$ }, 1, {0.00653595,  $\frac{1}{153}$ }}

## Palettes -> Basic Math Assistant

## Очистка рабочей области

In[331]:= **ClearAll[a]**

**ClearAll["Global`\*"]**

**a = .**

## Ввод чисел с другим основанием

---

В двоичной системе

In[334]:= **2<sup>101</sup>**

Out[334]= 5

В шестнадцатеричной системе

In[335]:= **16<sup>FF</sup>**

Out[335]= 255

Функция `BaseForm` преобразовывает число из одной системы в другую  
5 в десятичной системе преобразуется в двоичную

In[336]:= **BaseForm[5, 2]**

Out[336]/BaseForm= 101<sub>2</sub>

222, записанное в двоичной системе преобразуется в десятичную систему

In[337]:= **BaseForm[2<sup>11011110</sup>, 10]**

Out[337]/BaseForm= 222

`IntegerDigits`

In[338]:= **IntegerDigits[531]**

Out[338]= {5, 3, 1}

In[339]:= **N[1/153, 20]**

Out[339]= 0.0065359477124183006536

In[340]:= **RealDigits[1/153]**

Out[340]= {{ {6, 5, 3, 5, 9, 4, 7, 7, 1, 2, 4, 1, 8, 3, 0, 0} }, -2}

# Отложенное присвоение

---

```
In[341]:= a = c;  
          a := c;
```

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

```
In[343]:= a = c
```

```
Out[343]= c
```

```
In[344]:= c = 3;  
          a
```

```
Out[345]= 3
```

```
In[346]:= a := 2 c + 10  
          c = 1;  
          a
```

```
Out[348]= 12
```

```
In[349]:= c = 2;  
          a
```

```
Out[350]= 14
```

Еще один пример:

```
In[351]:= a := Random[ ]
```

```
In[352]:= a
```

```
Out[352]= 0.880576
```

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

```
In[353]:= a
```

```
Out[353]= 0.0424248
```

## Двумерная запись

In[354]:= **b := 2**

c[Ctrl][-]3:=6

In[355]:= **c<sub>3</sub> := 6**

In[356]:= **a<sub>3</sub>**

Out[356]:= 0.872073<sub>3</sub>

In[357]:= **a / 6**

Out[357]:= 0.120457

a[Ctrl][/]6

In[358]:=  **$\frac{a}{6}$**

Out[358]:= 0.154271

[Ctrl][2]a

In[359]:=  **$\sqrt{a}$**

Out[359]:= 0.438446

# Интервалы

---

```
In[360]:= mass = Interval[{0.9, 1.1}];
work = Interval[{9, 10}];
```

$$\text{velocity} = \sqrt{\frac{2 \text{ work}}{\text{mass}}}$$

```
Out[362]:= Interval[{4.0452, 4.71405}]
```

```
In[363]:= velocity[[1]]
```

```
Out[363]:= {4.0452, 4.71405}
```

$$\text{In[364]}:= \sqrt{2 * \frac{9}{1.1}}$$

```
Out[364]:= 4.0452
```

# Списки

---

## Одномерные

In[365]:= **a = {10, 20, 30, 40, 50}**

Out[365]:= {10, 20, 30, 40, 50}

In[366]:= **a[[2]]**

Out[366]:= 20

In[367]:= **a[[{3, 4}]]**

Out[367]:= {30, 40}

In[368]:= **Part[a, {3, 4}]**

Out[368]:= {30, 40}

In[369]:= **a[[2 ;; 4]]**

Out[369]:= {20, 30, 40}



# Списки

---

```
In[370]:= a = {10, 20, 30, 40, 50}
```

```
Out[370]:= {10, 20, 30, 40, 50}
```

```
In[371]:= First[a]
```

```
Out[371]:= 10
```

```
In[372]:= Last[a]
```

```
Out[372]:= 50
```

```
In[373]:= Rest[a]
```

```
Out[373]:= {20, 30, 40, 50}
```

```
In[374]:= Most[a]
```

```
Out[374]:= {10, 20, 30, 40}
```

```
In[375]:= Drop[a, 2]
```

```
Out[375]:= {30, 40, 50}
```

```
In[376]:= Select[a, 10 ≤ # ≤ 30 &]
```

```
Out[376]:= {10, 20, 30}
```

```
In[377]:= f[x_] = 10 ≤ x ≤ 30;
```

```
      Select[a, f]
```

```
Out[378]:= {10, 20, 30}
```

```
In[379]:=
```

# Многомерные списки

---

In[380]:= **a = {{10, 20, 30}, {40, 50, 60}, {80, 60, 30}, {41, 56, 67}}**

Out[380]:= **{{10, 20, 30}, {40, 50, 60}, {80, 60, 30}, {41, 56, 67}}**

In[381]:= **MatrixForm[a]**

Out[381]/MatrixForm= 
$$\begin{pmatrix} 10 & 20 & 30 \\ 40 & 50 & 60 \\ 80 & 60 & 30 \\ 41 & 56 & 67 \end{pmatrix}$$

In[382]:= **a[[2, 3]]**

Out[382]:= **60**

In[383]:= **MatrixForm[a[[{2, 3}, {1, 3}]]]**

Out[383]/MatrixForm= 
$$\begin{pmatrix} 40 & 60 \\ 80 & 30 \end{pmatrix}$$

In[384]:= **MatrixForm[a[[{2, 3}, 1 ;; 3]]]**

Out[384]/MatrixForm= 
$$\begin{pmatrix} 40 & 50 & 60 \\ 80 & 60 & 30 \end{pmatrix}$$

# Генерация списков

---

## Range

In[385]:= **Range[5]**

Out[385]= {1, 2, 3, 4, 5}

In[386]:= **Range[3, 5]**

Out[386]= {3, 4, 5}

In[387]:= **Subdivide[0, 1, 5]**

Out[387]=  $\left\{0, \frac{1}{5}, \frac{2}{5}, \frac{3}{5}, \frac{4}{5}, 1\right\}$

In[388]:=

# Генерация списков

---

## Table

```
In[389]:= Table[i2, {i, 2, 10}]
```

```
Out[389]:= {4, 9, 16, 25, 36, 49, 64, 81, 100}
```

```
In[390]:= Table[i2, {i, 2, 10, 2}]
```

```
Out[390]:= {4, 16, 36, 64, 100}
```

```
In[391]:= Table[{i, i2}, {i, 2, 10, 2}]
```

```
Out[391]:= {{2, 4}, {4, 16}, {6, 36}, {8, 64}, {10, 100}}
```

```
In[392]:= Table[10 i + j, {i, 1, 3}, {j, 2, 8, 2}]
```

```
Out[392]:= {{12, 14, 16, 18}, {22, 24, 26, 28}, {32, 34, 36, 38}}
```

# Генерация списков

---

In[393]:= **IdentityMatrix[3]**

Out[393]=  $\{\{1, 0, 0\}, \{0, 1, 0\}, \{0, 0, 1\}\}$

In[394]:= **Table[If[i == j, 1, 0], {i, 1, 3}, {j, 1, 3}]**

Out[394]=  $\{\{1, 0, 0\}, \{0, 1, 0\}, \{0, 0, 1\}\}$

In[395]:= **ConstantArray[3, 5]**

Out[395]=  $\{3, 3, 3, 3, 3\}$

In[396]:= **Table[3, {5}]**

Out[396]=  $\{3, 3, 3, 3, 3\}$

## Преобразование списков

---

In[397]:= **a = Table**[ $i^2$ , {i, 2, 10, 2}]

Out[397]:= {4, 16, 36, 64, 100}

In[398]:= **Reverse**[a]

Out[398]:= {100, 64, 36, 16, 4}

In[399]:= **Join**[{1, 2}, {3, 4}]

Out[399]:= {1, 2, 3, 4}

In[400]:= **{Min[a], Max[a]}**

Out[400]:= {4, 100}

In[401]:= **{a + a, a \* a, a / a}**

Out[401]:= {{8, 32, 72, 128, 200}, {16, 256, 1296, 4096, 10000}, {1, 1, 1, 1, 1}}

In[402]:= **a = {3, 4};**

**b = {1, 2};**

**a<sup>b</sup>**

Out[404]:= {3, 16}

## Добавление и удаление элементов

---

```
In[405]:= a = Table[i^2, {i, 2, 10, 2}]
```

```
Out[405]:= {4, 16, 36, 64, 100}
```

```
In[406]:= {Append[{1, 2, 3}, 30], Prepend[{1, 2, 3}, 30]}
```

```
Out[406]:= {{1, 2, 3, 30}, {30, 1, 2, 3}}
```

```
In[407]:= {Insert[{1, 2, 3, 4}, 17, 2], Insert[{1, 2, 3, 4}, 17, -2]}
```

```
Out[407]:= {{1, 17, 2, 3, 4}, {1, 2, 3, 17, 4}}
```

```
In[408]:= RotateLeft[a]
```

```
Out[408]:= {16, 36, 64, 100, 4}
```

```
In[409]:= RotateRight[a]
```

```
Out[409]:= {100, 4, 16, 36, 64}
```

# Сортировка

---

```

In[410]:= a = RandomInteger[{-10, 10}, 4]
Sort[a]
f[a_, b_] := a > b;
Sort[a, f]
Sort[a, #1 > #2 &]

Out[410]= {10, 3, -6, 5}
Out[411]= {-6, 3, 5, 10}
Out[413]= {10, 5, 3, -6}
Out[414]= {10, 5, 3, -6}

In[415]:= data = {{x, 2}, {y, 1}, {z, 5}};
In[416]:= Sort[data, #1[[2]] < #2[[2]] &]
Out[416]= {{y, 1}, {x, 2}, {z, 5}}

In[417]:= a
Ordering[a]

Out[417]= {10, 3, -6, 5}
Out[418]= {3, 2, 4, 1}

```



# Матрицы

## Изменение/замена строк и столбцов

```
In[419]:= a = Table[0, {5}, {5}];
```

```
a // MatrixForm
```

Out[420]/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[421]:= a[[All, 2]] = 1;
```

```
a[[;;, 2]] = 1;
```

```
a // MatrixForm
```

Out[423]/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[424]:= a[[All, 2]] = Range[5];
```

```
a // MatrixForm
```

Out[425]/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[426]:= a = Table[0, {5}, {5}];
```

$$a[[1 ;; 2, 3 ;; 4]] = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix};$$

```
a // MatrixForm
```

Out[428]/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[429]:= AppendTo[a, {7, 8, 9, 10, 11}]
```

```
a // MatrixForm
```

```
Out[429]= {{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[430]/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[431]:= Do[AppendTo[a[[i]], 9], {i, 1, 6}]
```

```
a // MatrixForm
```

Out[432]/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[433]:= Partition[{a, b, c, d, e, f}, 2]
```

```
Out[433]:= {{{{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[434]:= Partition[{a, b, c, d, e, f}, 2, 1]
```

```
Out[434]:= {{{{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[435]:= Tuples[{1, 2, 3}, 2]
```

```
Out[435]:= {{1, 1}, {1, 2}, {1, 3}, {2, 1}, {2, 2}, {2, 3}, {3, 1}, {3, 2}, {3, 3}}
```

## Subsets

```
In[436]:= Subsets[{1, 2, 3}, 2]
```

```
Out[436]:= {{}, {1}, {2}, {3}, {1, 2}, {1, 3}, {2, 3}}
```

```
In[437]:= Subsets[{1, 2, 3}, {2}]
```

```
Out[437]:= {{1, 2}, {1, 3}, {2, 3}}
```

```
In[438]:= Subsets[{{x1, y1}, {x2, y2}, {x3, y3}, {x4, y4}, {x5, y5}}, {3}]
```

```
Out[438]:= {{{{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[439]:= a = {1, 2, 3.0, 4, 5.0, 6, 7, {1, 3}};
```

```
In[440]:= Count[a, _List]
```

```
Out[440]= 1
```

```
In[441]:= Count[a, _Integer]
```

```
Out[441]= 5
```

```
In[442]:= Count[a, a_ /; a > 5 || a < 2]
```

```
Cases[a, a_ /; a > 5 || a < 2]
```

```
Out[442]= 3
```

```
Out[443]= {1, 6, 7}
```

```
In[444]:= a = {"fd", 1, 2, 3, "4"}
```

```
Position[a, _String]
```

```
Cases[a, _String]
```

```
Out[444]= {fd, 1, 2, 3, 4}
```

```
Out[445]= {{1}, {5}}
```

```
Out[446]= {fd, 4}
```

# Cases

---

## Cases

```

In[447]:= Count[{y3, 1, 2, 3, x-1, x, x2, x3}, a_b_/; b == -1]
          Position[{y3, 1, 2, 3, x-1, x, x2, x3}, a_b_/; b == 2]
          Cases[{y3, 1, 2, 3, x-1, x, x2, x3}, a_b_/; b == 2]

Out[447]= 1

Out[448]= { {7} }

Out[449]= { x2 }

In[450]:= Table[Random[Real, {0, 100}], 3], {10}]
          Count[%, a_ /; a < 30]

Out[450]= {38.8, 40.4, 46.4, 16.4, 22.6, 80.3, 94.7, 87.5, 32.3, 26.6}

Out[451]= 3

```

## Select

```

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

Out[ ]:= {4, 5}

```

## Векторная алгебра

---

```

In[452]:= a = {1, 2, 3};
          b = {3, 4, 5};

In[454]:= a.b
Out[454]= 26

In[455]:= Cross[a, b]
Out[455]= {-2, 4, -2}

In[456]:= Norm[a]
Out[456]=  $\sqrt{14}$ 

In[457]:= Projection[a, b]
Out[457]=  $\left\{ \frac{39}{25}, \frac{52}{25}, \frac{13}{5} \right\}$ 

In[458]:= 
$$\frac{a.b}{\text{Norm}[b]} \frac{b}{\text{Norm}[b]}$$

Out[458]=  $\left\{ \frac{39}{25}, \frac{52}{25}, \frac{13}{5} \right\}$ 

```

## Функции

---

### Объявление функций

Встроенные функции Математики начинаются с большой буквы

```

In[459]:= {Sin[2], N[Sin[2]], Sin[2.0]}
Out[459]= {Sin[2], 0.909297, 0.909297}

In[460]:= hyp = Function[{x, y},  $\sqrt{x^2 + y^2}$ ]
Out[460]= Function[{x, y},  $\sqrt{x^2 + y^2}$ ]

In[461]:= hyp[1, 3]
Out[461]=  $\sqrt{10}$ 

```

```
In[462]:= Clear[hyp]
```

```
hyp[x_, y_] =  $\sqrt{x^2 + y^2}$ 
```

```
Out[463]=  $\sqrt{x^2 + y^2}$ 
```

x\_ -- некоторое значение, которое мы будем называть при определении функции “x”

```
In[464]:= hyp[x_, y_] :=  $\sqrt{x^2 + y^2}$ 
```

# Анонимная функция

---

Через анонимную функцию

```
In[465]:= hyp =  $\sqrt{\#1^2 + \#2^2}$  &;  
hyp[1, 2]
```

Out[466]=  $\sqrt{5}$

Пример (почему это может быть удобно)

```
In[467]:=  $\sqrt{2}$  // N[#, 10] &
```

Out[467]= 1.414213562

Sort, Select

In[468]:=

In[469]=



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



# Постфиксная, префиксная и инфиксная запись

---

## Постфиксная запись

In[470]:= **12 // Sin**

Out[470]:= **Sin[12]**

In[471]:=  **$\sqrt{2}$  // N[#, 10] &**

Out[471]:= **1.414213562**

## Инфиксная запись

In[472]:= **2 ~ Plus ~ 1**

Out[472]:= **3**

## Префиксная запись

In[473]:= **Sin@d**

Out[473]:= **Sin[d]**

## Функциональная запись

In[474]:= **Sin[1.2]**

Out[474]:= **0.932039**

## Функции с индексами

---

### Функции с индексами

```
In[564]:= ClearAll[f]
          ClearAll[f]

In[566]:= f[1][x_, y_] = x + y;
          f[2][x_, y_] = x - y;

In[569]:= {f[1][1, 2], f[2][1, 2]}

Out[569]= {3, -1}

In[553]:= f1[x_, y_] = x + y;
          f2[x_, y_] = x - y;
          fRandomInteger[{1,2}][1, 2]

Out[555]= 3
```

## Значения по умолчанию

---

```
In[481]:= f[x_, y_ : 10] := 10 x + y;  
          f[10]  
          f[10, 20]
```

```
Out[482]= True
```

```
Out[483]= False
```

# Кусочные функции

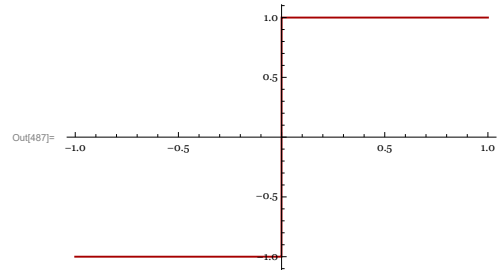
---

```
In[484]:= f[x_] := If[x > 0, 1, -1]
```

```
In[485]:= g[x_] := 1 /; x > 0
```

```
g[x_] := -1 /; x ≤ 0
```

```
Plot[g[x], {x, -1, 1}]
```



# Замены

---

## Замена

In[488]:=  $d + 3 + \sqrt{h + 1}$

Out[488]:=  $3 + d + \sqrt{1 + h}$

In[489]:=  $d + 3 + \sqrt{h + 1} /. \{d \rightarrow 3, h \rightarrow 5\}$

Out[489]:=  $6 + \sqrt{6}$

Показать в приближенном виде:

In[490]:=  $N[d + 3 + \sqrt{h + 1} /. \{d \rightarrow 3, h \rightarrow 5\}]$

Out[490]:= 8.44949

In[491]:=  $\{a1, b1, c1, d1\} /. a1 \rightarrow 10$

Out[491]:=  $\{10, b1, c1, d1\}$

In[492]:=  $\{a1, b1, c1, d1\} /. \{a1 \rightarrow 10, c1 \rightarrow 20\}$

Out[492]:=  $\{10, b1, 20, d1\}$

## Отложенная замена

---

```
In[493]:= {a1, a1, a1} /. a1 -> RandomReal[]  
          {a1, a1, a1} /. a1 :> RandomReal[]
```

```
Out[493]= {0.54926, 0.54926, 0.54926}
```

```
Out[494]= {0.797451, 0.624844, 0.193939}
```

```
In[495]:=
```

## Функции и списки

---

In[496]:= **q = {1, 2, 3, 4, 5}**

**Sin[q]**

Out[496]:= {1, 2, 3, 4, 5}

Out[497]:= {Sin[1], Sin[2], Sin[3], Sin[4], Sin[5]}

In[498]:= **f[x\_] := x<sup>2</sup> + 3 \* Sin[x]**

**f[q]**

Out[499]:= {1 + 3 Sin[1], 4 + 3 Sin[2], 9 + 3 Sin[3], 16 + 3 Sin[4], 25 + 3 Sin[5]}

In[500]:= **f = .**

**f[x\_] := Total[x]**

**f[q]**

Out[502]:= 15

In[503]:= **Map[f, {1, 2, 3}]**

Out[503]:= {1, 2, 3}

In[504]:= **f /@ {1, 2, 3}**

Out[504]:= {1, 2, 3}

# Apply

---

```
In[505]:= Apply[f, {1, 2, 3}]
          f@@{1, 2, 3}
```

```
Out[505]= f[1, 2, 3]
```

```
Out[506]= f[1, 2, 3]
```

## Apply для первого уровня

```
In[507]:= Apply[f, {{1, 2}, {2, 3}, {3, 4}}, {1}]
          f@@@{{1, 2}, {2, 3}, {3, 4}}
```

```
Out[507]= {False, False, False}
```

```
Out[508]= {False, False, False}
```

```
In[509]:= Apply[ss[#2, #1] &, {{1, 2}, {2, 3}, {3, 4}}, {1}]
          ss[#2, #1] &@@@{{1, 2}, {2, 3}, {3, 4}}
```

```
Out[509]= {ss[2, 1], ss[3, 2], ss[4, 3]}
```

```
Out[510]= {ss[2, 1], ss[3, 2], ss[4, 3]}
```

```
In[511]:= 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[511]= Total[{ {1, 2}, {3, 4}, {5, 6, 7} }]
```

```
In[512]:= Total[{{1, 2}, {3, 4}, {5, 6}}]
```

```
Out[512]= {9, 12}
```

```
In[513]:= #12 + #2 * 10 &@@@{{1, 2}, {3, 4}, {5, 6}}
```

```
Out[513]= {21, 49, 85}
```



# Map

---

```
In[514]:= Map[Apply[Plus, #] &, {{1, 2}, {3, 4}, {5, 6, 7}}]
```

```
Out[514]= {3, 7, 18}
```

```
In[515]:= Apply[Plus, #] & /@ {{1, 2}, {3, 4}, {5, 6, 7}}
```

```
Out[515]= {3, 7, 18}
```

# MapThread

---

```
In[518]:= MapThread[fun, {{p1, p2, p3}, {1, 2, 3}}]
```

```
Out[518]:= {fun[p1, 1], fun[p2, 2], fun[p3, 3]}
```

```
In[519]:= MapThread[fun, {{p1, p2, p3}, {1, 2, 3}, {10, 20, 30}}]
```

```
Out[519]:= {fun[p1, 1, 10], fun[p2, 2, 20], fun[p3, 3, 30]}
```

## Центр масс системы точек

```

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

In[5]:= Total[MapThread[#1 * #2 &, {r, m}]] / Total[m]

Out[5]= {19/7, 26/7}

In[6]:= Total[r]

Out[6]= {9, 16}

In[7]:= Total[r, 2]

Out[7]= 25

In[8]:= Total[r, {2}]

Out[8]= {3, 5, 10, 7}

```

### Вариант 2

```

In[581]:= Transpose[{r, m}]

Out[581]= {{ {1, 2}, 1}, {{3, 2}, 3}, {{4, 6}, 2}, {{1, 6}, 1}}

In[583]:= Total[Map[#[[1]] * #[[2]] &, Transpose[{r, m}]]] / Total[m]

Out[583]= {19/7, 26/7}

```

### Вариант 3

```

In[585]:= Total[r * m] / Total[m]

Out[585]= {19/7, 26/7}

```

### Код в процедурном стиле

```
In[578]:= rc = 0;
For[i = 1, i ≤ Length[r], i++,
  rc = rc + (r[[i]] * m[[i]]) / Total[m]
]
rc
```

```
Out[580]:= { $\frac{19}{7}$ ,  $\frac{26}{7}$ }
```

## Расстояние между парами точек

---

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

```
In[586]:= r = {{1, 2}, {3, 2}, {4, 6}, {1, 6}};
```

```
pairs = Subsets[r, {2}]
```

```
Out[587]:= {{{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[588]:= pairs[[1]]
```

```
Out[588]:= {{1, 2}, {3, 2}}
```

```
In[589]:= Apply[Subtract, pairs[[1]]]
```

```
Out[589]:= {-2, 0}
```

```
In[590]:= Apply[Subtract, pairs[[1]]]^2
```

```
Out[590]:= {4, 0}
```

```
In[591]:= Sqrt[Total[Apply[Subtract, pairs[[1]]]^2]]
```

```
Out[591]:= 2
```

## Расстояние между парами точек

```
In[592]:= d = Map[Sqrt[Total[Apply[Subtract, #] ^2]] &, pairs] // N
Ordering[d]
```

```
Out[592]:= {2., 5., 4., 4.12311, 4.47214, 3.}
```

```
Out[593]:= {1, 6, 3, 4, 5, 2}
```

```
In[594]:= d = Map[Sqrt[Total[Apply[Subtract, #] ^2]] &, pairs]
Ordering[d]
```

```
Out[594]:= {2, 5, 4,  $\sqrt{17}$ ,  $2\sqrt{5}$ , 3}
```

```
Out[595]:= {1, 6, 3, 2, 5, 4}
```

```
In[596]:= Max[d]
```

```
Out[596]:= 5
```

```
In[597]:= pairs[[Ordering[d][[-1]]]
```

```
Out[597]:= {{3, 2}, {4, 6}}
```

```
In[598]:= TreeForm[d]
```

