

## Практическая работа № 1

### Вычисление простого выражения

$$\sqrt{100}, |-10|, 10! = \sqrt{-10}.$$

### Определение переменных и функций.

1) Определить переменные  $a = 3.4$ ,  $b = 6.22$ ,  $c = 0.149$  и выражения:

$$Z := \frac{2ab + \sqrt[3]{c}}{\sqrt{(a^2 + b^{a+c}) \cdot c}},$$

$$N := e^{\sin c} \cos \frac{a}{b}$$

2) Определить  $\operatorname{pr}(x, y) = 2 \cdot \sin(x-y) \cdot a - 4$

$$\operatorname{pr}(5\pi/2, \pi) =$$

3) Выполнить следующие операции с комплексными числами:

а)  $Z := -3 + 2i$ ,  $|Z| =$ ,  $\operatorname{Re}(Z) =$ ,  $\operatorname{Im}(Z) =$ ,  $\arg(Z) =$ ,

$$\sqrt{\bar{Z}} =$$
,  $2Z =$

б)  $Z1 := 1 + 2i$ ,  $Z2 := 3 + 4i$ ,

$$Z1 + Z2 =$$
,  $Z1 - Z2 =$ ,  $Z1 \cdot Z2 =$ ,  $Z1/Z2 =$ .

### Определение и использование ранжированного аргумента:

1) Выполнить следующие операции:

$$i := 1 \dots 0, \quad \sum_i i =$$
,  $\prod_i i =$

2) Дано целое положительное число  $n$ . Вычислить  $(1 + \frac{1}{1^2}) + (1 + \frac{1}{2^2}) + (1 + \frac{1}{3^2}) + \dots + (1 + \frac{1}{n^2})$

### Использование условного оператора:

1. Дано действительное число  $x$ . Вычислить  $f(x)$ , если

$$f(x) = \begin{cases} 0 & \text{при } x \leq 0, \\ x^2 - x & \text{при } 0 < x \leq 1, \\ x^2 - \sin \pi x^2 & \text{при других } x. \end{cases}$$

### Построение графиков

1) Построить графики

$$y = 0.75 \cos x, \quad y = 0.25 \sin 3x, \quad y = \cos^3 x$$

а) без указания диапазона изменения аргумента

б)  $x \in [-4; 4]$

2) Построить декартовы и полярные графики следующих функций:

$$X(\alpha) := \cos(\alpha) \cdot \sin(\alpha)$$

$$Y(\alpha) := 1.5 \cos(\alpha)^2 - 1$$

$$P(\alpha) := \cos(\alpha)$$

Для этого необходимо определить  $\alpha$  как дискретный аргумент на интервале от 0 до  $2\pi$  с

шагом  $\pi/30$ .

3) Построить график

$z = x * y$ , где  $x, y \in [-3; 3]$ ,  $z \in [-4; 4]$

**Нахождение корней уравнения:**

Найти корень уравнения  $3ax - 7x^2 + x^3 = 0$ . Проверить результат графически

## Практическая работа № 2

### Вычисление простого выражения

$$\sqrt{100}, |-10| =, \sqrt{-10}.$$

### Определение переменных

1) Определить переменные  $a = 3.4$ ,  $b = 6.22$ ,  $c = 0.149$  и выражения:

$$Z = \frac{2ab + \sqrt[3]{c}}{\sqrt{(a^2 + b^{a+c}) \cdot c}},$$

$$N = e^{\sin c} \cos \frac{a \cdot \pi}{b}$$

### Построение графиков

1) Построить графики

$$y = 0.75 \cos x, y = 0.25 \sin 3x, y = \cos^3 x$$

$x \in [-4; 4]$  с шагом 0.1

2) Построить декартовы и полярные графики следующих функций:

$$X(\alpha) := \cos(\alpha) \cdot \sin(\alpha)$$

$$Y(\alpha) := 1.5 \cos(\alpha)^2 - 1$$

$$P(\alpha) := \cos(\alpha)$$

Для этого необходимо определить  $a$  как дискретный аргумент на интервале от  $0$  до  $2\pi$  с шагом  $\pi/30$ .

Для построения графиков будет полезна следующая информация.

### Функция plot2d

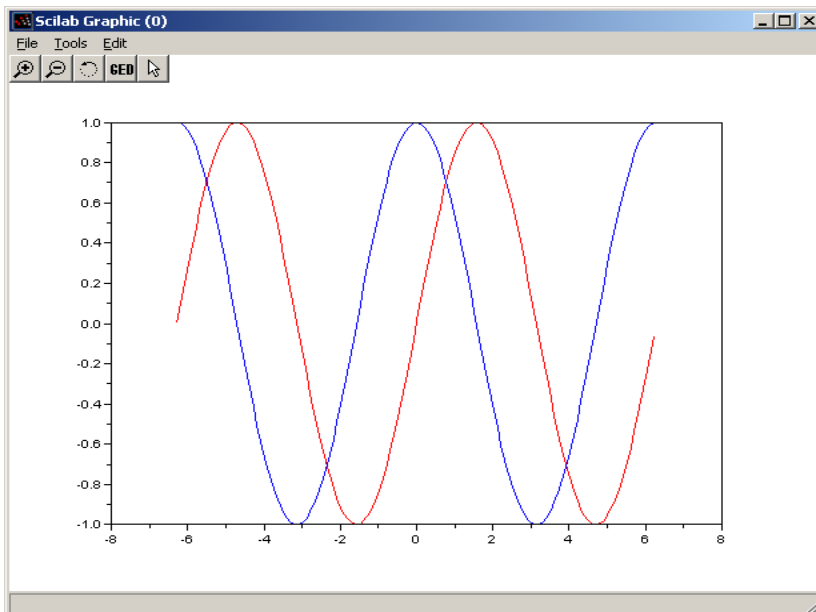
В общем виде обращение к функции имеет вид:

```
plot2d([loglog], x, y, [key1=value1, key2=value2, ..., keyn=valuen])
```

- `logflag` – строка из двух символов, каждый из которых определяет тип осей ( $n$  – нормальная ось,  $l$  – логарифмическая ось), по умолчанию `"nn"`;
- `x` – массив абсцисс;
- `y` – массив ординат (или матрица, каждый столбец которого содержит массив ординат очередного графика) (количество элементов в массиве `x` и `y` должно быть одинаковым), если `x` и `y` – являются матрицами одного размера, то в этом случае, каждый столбец матрицы `y` отображается относительно соответствующего столбца матрицы `x`;
- `keyi=valuei` – последовательность значений параметров графиков, возможны следующие значения параметров: `style` – определяет массив (`mas`) числовых значений цветов графика (`id` цвета), количество элементов массива совпадает с количеством изображаемых графиков, по умолчанию, по умолчанию представляет собой массив `masi=i`, цвет  $i$ -й линии совпадает с номером  $i$ , для формирования `id` соответствующего цвета (кода цвета) можно воспользоваться функцией `color`, которая по названию (`color("цвет")`) или коду `grb` (`color(r, g, b)`) цвета формирует нужный `id` (код) цвета. Если значение стиля отрицательное то это будет точечный график без соединения точек между собой линиями. Пример построения

нескольких графиков различного цвета приведен ниже

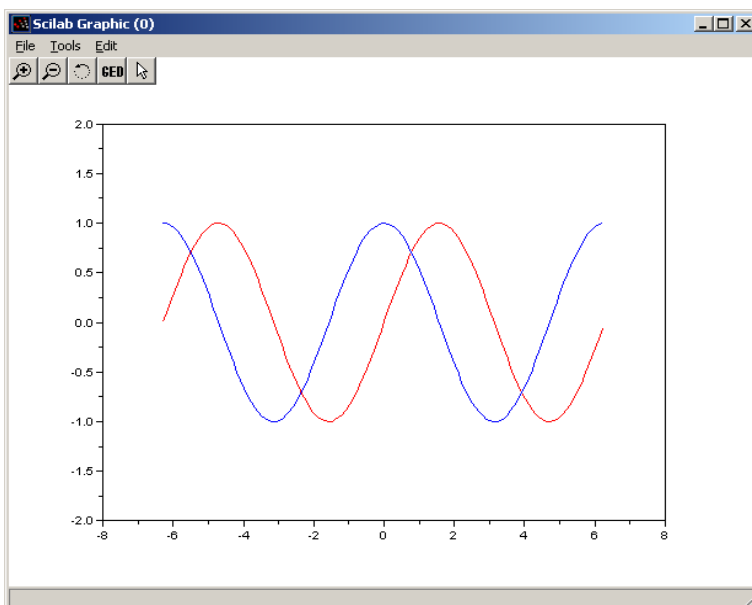
```
x=[-2*%pi:0.1:2*%pi];  
y=[sin(x); cos(x)];  
plot2d(x,y',style=[color("red"), color("blue")]);
```



Использование параметра style в функции plot2d

- rect – этот вектор [xmin, ymin, xmax, ymax] определяет размер окна вокруг графика.

```
x=[-2*%pi:0.1:2*%pi];  
y=[sin(x); cos(x)];  
plot2d(x,y',style=[color("red"), color("blue")],rect=[-8,-2,8,2]);
```



Использование параметра rect

- frameflag – параметр определяет окно в котором, будет изображаться график, он может принимать следующие значения: 0 – не вычислять размеры окна, использовать значения по умолчанию или значения из предыдущего графика, 1 – размер окна

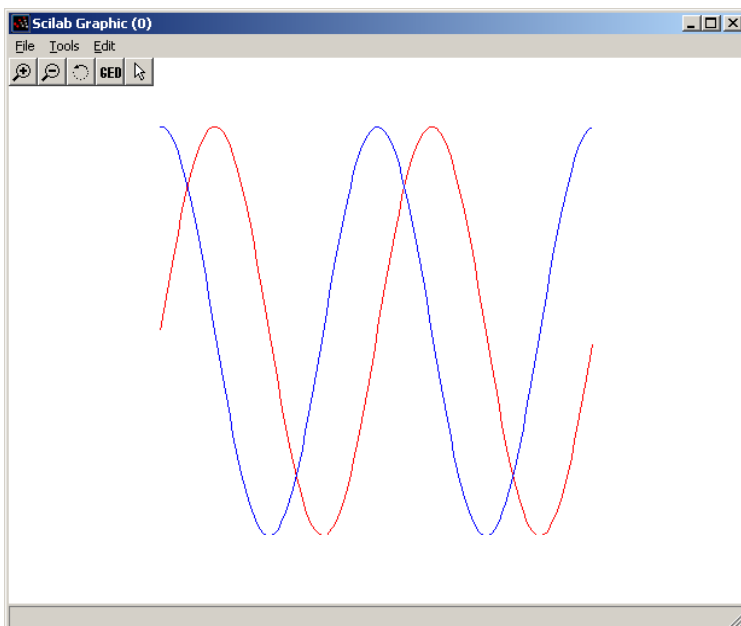
определяется параметром `rect`, 2 – размер окна определяется из соотношения между минимальным или максимальным значениями  $x$  и  $y$ , 3 – размер окна определяется параметром `rect` в в изометрическом масштабе, 4– размер окна определяется из соотношения между минимальным или максимальным значениями  $x$  и  $y$  в изометрическом масштабе,

- `axesflag` - параметр, который определяет рамку вокруг графика, следует выделить следующие значения этого параметра: 0 – нет рамки вокруг графика (; 1 – изображение рамки, ось  $y$  слева ; 3 – изображение рамки, ось  $y$  справа (см. рис. 4.11); 5 – изображение осей проходящих через точку  $(0,0)$  (см. следующие рисунки);

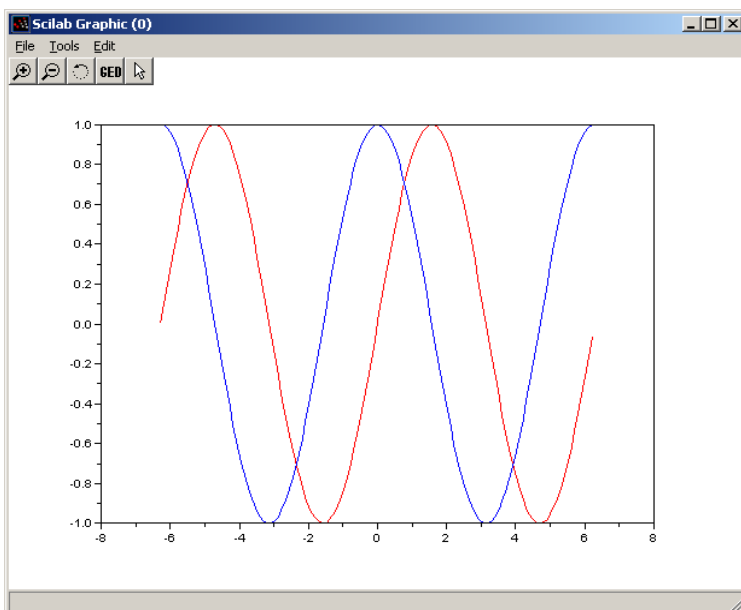
```
x=[-2*%pi:0.1:2*%pi];
```

```
y=[sin(x); cos(x)];
```

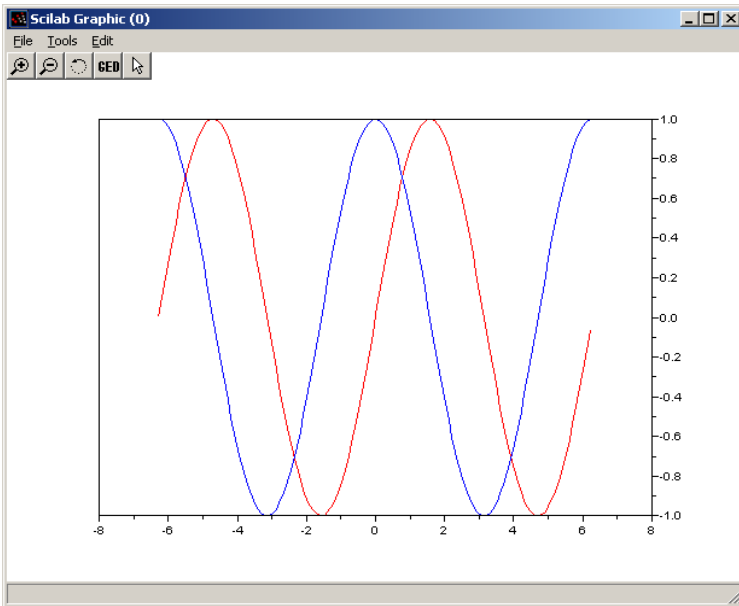
```
plot2d(x,y',style=[color("red"), color("blue")], axesflag=0);
```



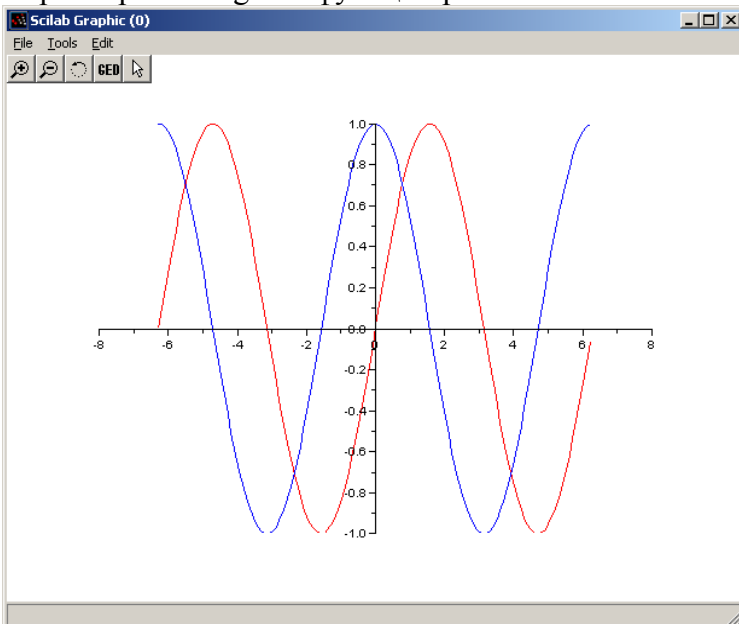
Параметр `axesflag=0` в функции `plot2d`



Параметр `axesflag=1` в функции `plot2d`



Параметр axesflag=3 в функции plot2d



Параметр axesflag=5 в функции plot2d

- `max` – этот параметр используют, если параметр `axesflag` равен 1, `max` представляет массив из четырех значений `[nx, Nx, ny, Ny]` – где `Nx` (`Ny`) – число основных делений с подписями под осью `X` (`Y`), `nx` (`ny`) – число промежуточных делений;
- `leg` – строка, определяющая легенды для каждого графика, структура строки такая: `"leg1@leg2@leg3@...@legn"`, где `leg1` – легенда первого графика, ..., `legn` – легенда первого графика.

Пример построения графиков функций с использованием параметра `max` при построении функции `plot2d`.

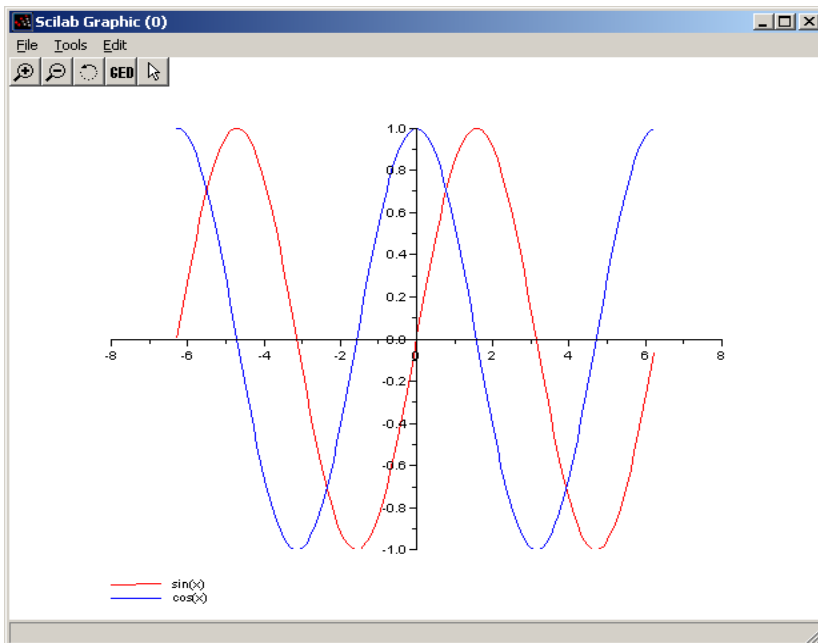
```
x=[-8:0.1:8];
```

```
y=[sin(x); cos(x)];
```

```
plot2d(x,y', style=[color("red"), color("blue")], axesflag=1,
max=[4, 9, 3, 6]);
```

Пример построения графиков функции с использование легенд.

```
x=[-2*%pi:0.1:2*%pi];
y=[sin(x); cos(x)];
plot2d(x,y',style=[color("red"), color("blue")], axesflag=5,
leg="sin(x)@cos(x)");
```



Использование параметров `leg` и `axesflag` в функции `plot2d`

Функцию `plot2d` можно использовать для построения точечных графиков. В этом случае обращение к функции имеет вид

```
plot2d(x, y, d),
```

здесь `d` – отрицательное число, определяющее тип маркера

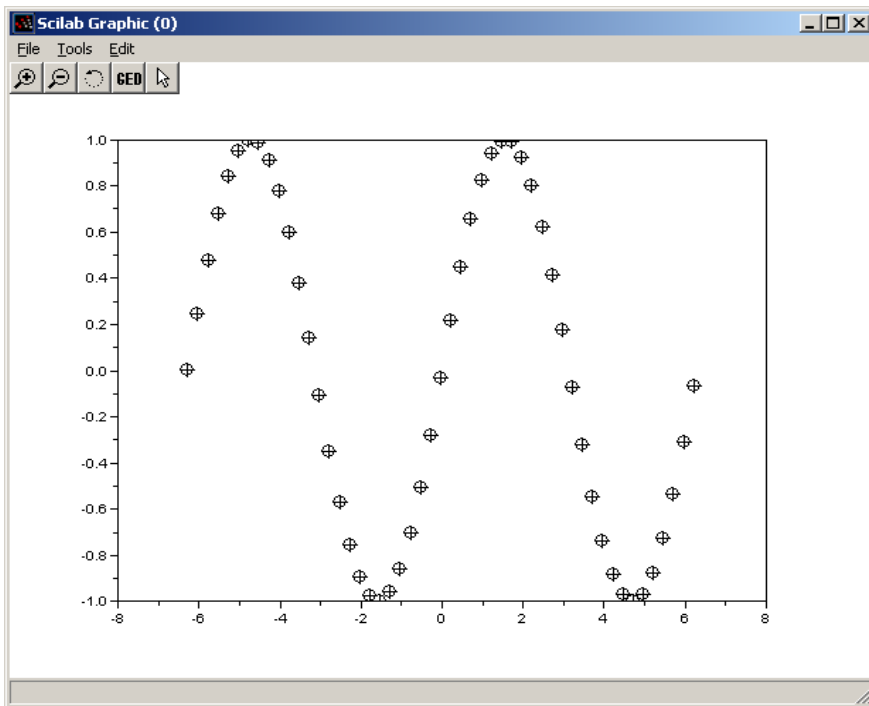
```
x=[-2*%pi:0.25:2*%pi];
```

```
y=sin(x);
```

```
plot2d(x, y, -3);
```

Изображение сетки на графике

Для изображения сетки следует воспользоваться функцией `xgrid(color)`, `color` определяет `id` цвета линии сетки.



Построение точечного графика

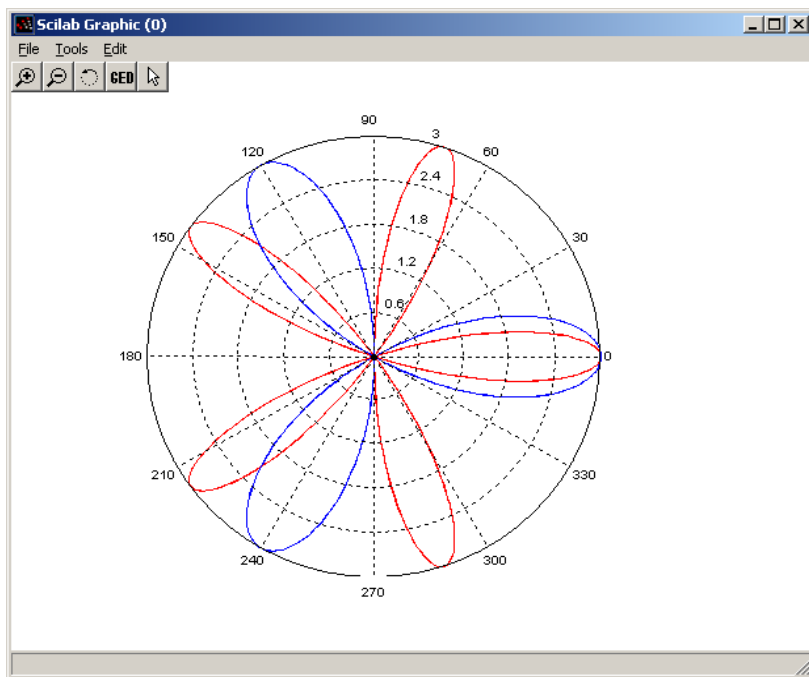
### Построение полярных графиков

Для построения графиков в полярной системе координат в Scilab служит функция `polarplot`  
`polarplot(fi, ro, [key1=value1, key2=value2, ..., keyn=valuen])`  
 Здесь  $fi$  - полярный угол,  $ro$  – полярный радиус.

Рассмотрим пример построения полярных графиков  $\rho = 3\cos(5\varphi)$ ,  $\rho_1 = 3\cos(3\varphi)$

```
fi=0:0.01:2*%pi;
ro=3*cos(5*fi);
ro1=3*cos(3*fi);
polarplot(fi,ro,style=color("red"));
polarplot(fi,ro1,style=color("blue"));
```





Пример построения полярных графиков

## Практическая работа №3

### Задания на практическую работу:

1. Написать программу, формирующую матрицу случайных чисел (число строк и столбцов матрицы вводит пользователь) и производящую следующие действия над матрицей:
  - вывод матрицы;
  - вывод максимального и минимального элементов матрицы;
  - вывод суммы и произведения элементов;
  - вывод суммы и произведения элементов, расположенных на главной диагонали;
  - вывод обратной матрицы.
2. Написать программу, формирующую одномерный массив, значение каждого элемента которого равно квадрату значения индекса.
3. Написать программу, вычисляющую факториал заданного числа ( $n! = 1*2*3*...*n$ ) двумя способами: с помощью цикла с параметром и с помощью цикла с условием.

Программы должны быть написаны в Scilab и быть оформлены в виде отдельных файлов для возможности многократного использования.