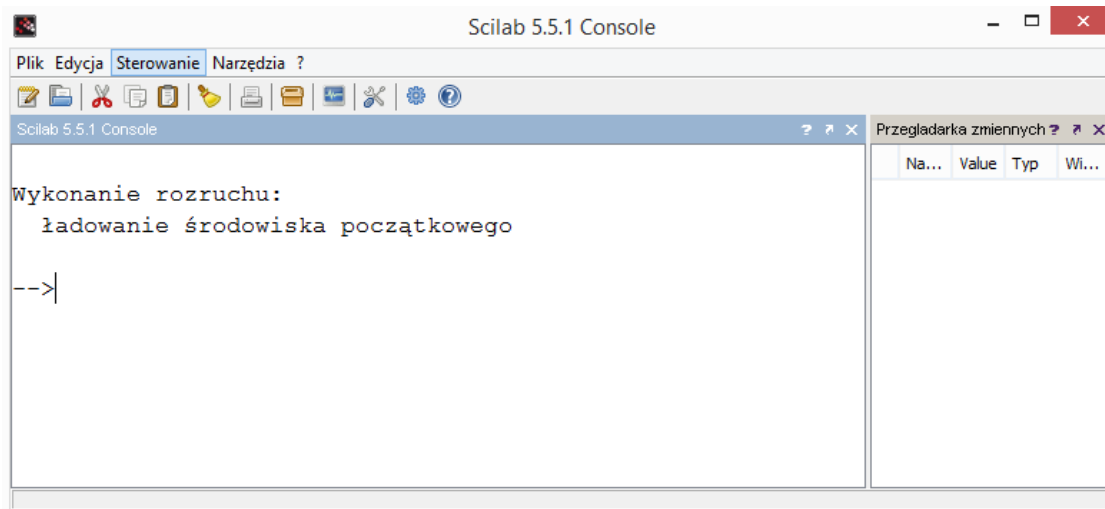


SCILAB

Po uruchomieniu programu SCILAB na ekranie ukaże się następujące okno:



Rys01 – Okno główne programu SCILAB (okno konsolowe)

W powyższym oknie można wykonywać wszystkie polecenia programu SCILAB oraz wyświetlać dane zapisane w zmiennych.

Podstawowe operacje:

Definicja zmiennej x z przypisaniem wartości 4:

`x=4`

Definicja macierzy X wypełnionej zerami o 3

wierszach i 2 kolumnach:

`X=zeros(3,2)`

Definicja macierzy X wypełnionej jedynkami o 5

wierszach i 3 kolumnach:

`A=ones(5, 3)`

Przypisanie do elementu w pierwszym wierszu i pierwszej kolumnie macierzy X wartości 6:

`A(1,1)=6`

Pobranie wartości z drugiego wiersza i trzeciej kolumny macierzy X do zmiennej y:

`y=A(2,3)`

```
-->x=4
x =

    4.

-->X=zeros(3,2)
X =

    0.    0.
    0.    0.
    0.    0.
```

```
-->A=ones(5,3)
A =

    1.    1.    1.
    1.    1.    1.
    1.    1.    1.
    1.    1.    1.
    1.    1.    1.

-->A(1,1)=6
A =

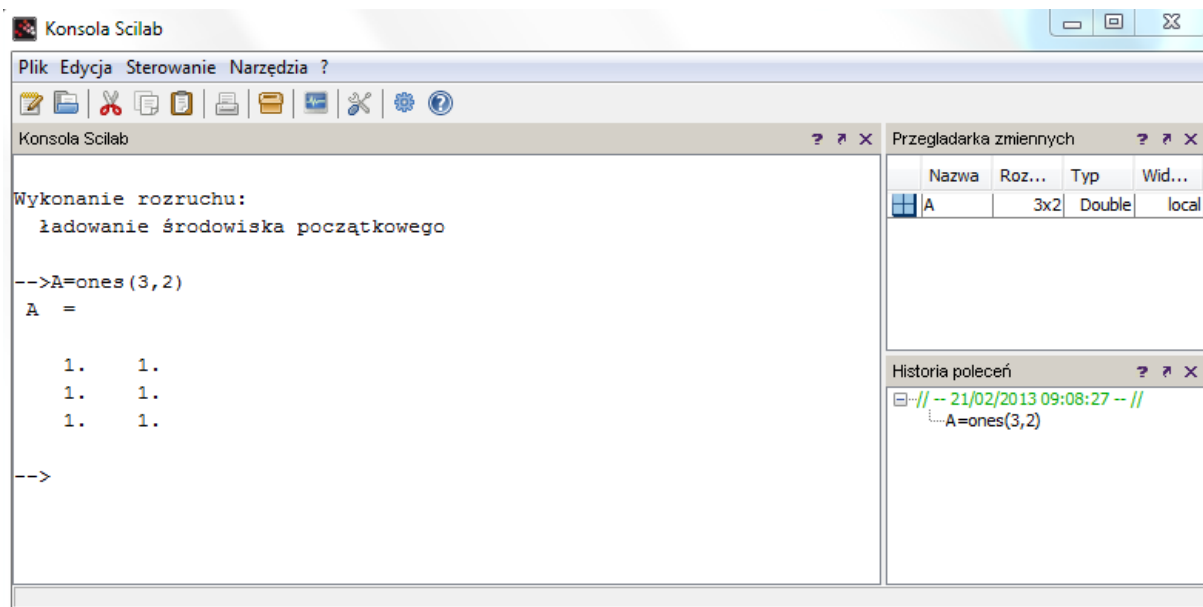
    6.    1.    1.
    1.    1.    1.
    1.    1.    1.
    1.    1.    1.
    1.    1.    1.

-->y=A(2,3)
y =

    1.
```

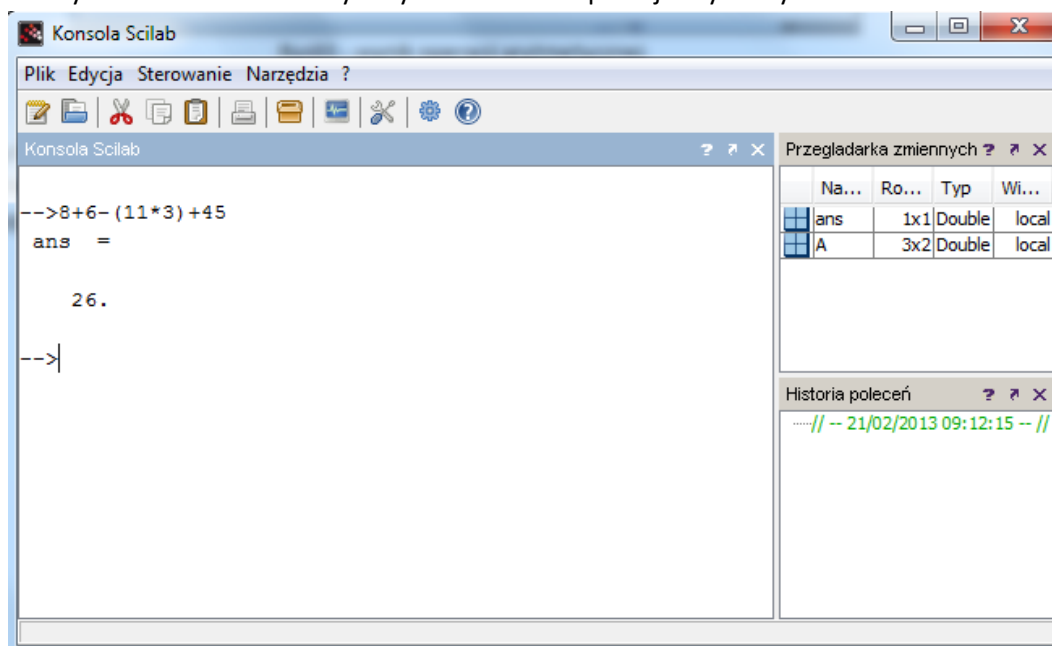
W przypadku, gdy na końcu linii nie wstawimy średnika wówczas wynik tej operacji zostanie wyświetlony na ekranie.

Dla przykładu na Rys02 przedstawiono utworzenie macierzy X o 3 wierszach i 2 kolumnach:



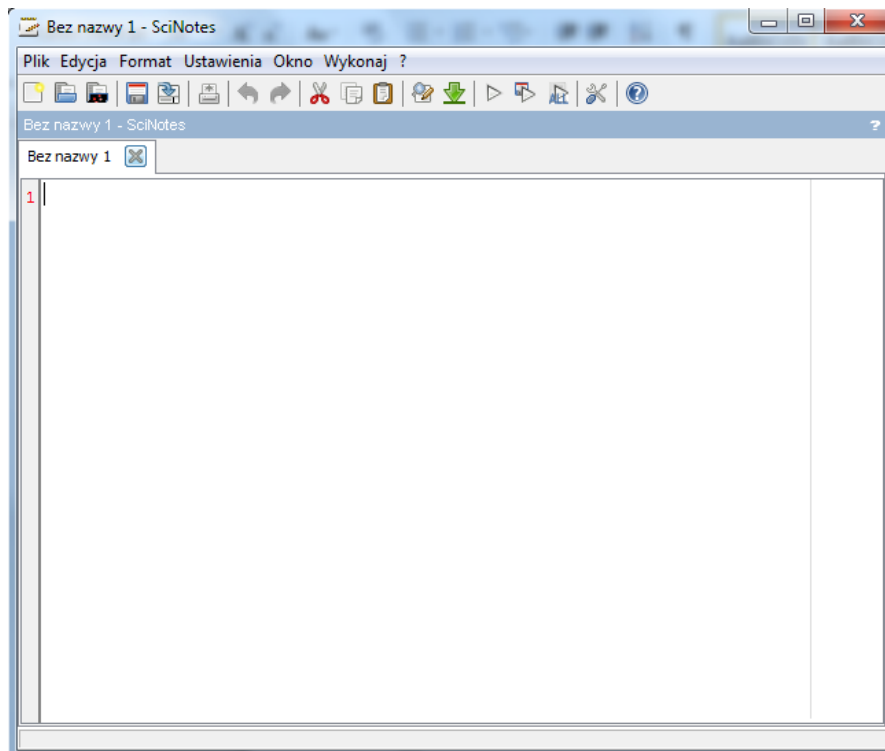
Rys02 – Przykład użycia instrukcji ones

W oknie z Rys01 można również wykonywać wszelkie operacje arytmetyczne:



Rys03 - wynik operacji arytmetycznej

Aby tworzyć programy (tzw. skrypty) należy wybrać opcję Narzędzia|SciNotes z górnego menu. Wówczas uruchomi się edytor programu SCILAB (patrz Rys04), w którym użytkownik może pisać kod programu.



Rys04 – Edytor programu SCILAB

Przykład 1

Poniżej przedstawiono program pobierający od użytkownika dwie liczby a następnie obliczający sumę tych liczb i wyświetlający ją na ekranie (patrz Rys05).

```
c01_p1.sce
1 //pobranie pierwszej liczby A
2 A=input('Podaj A: ');
3 //pobranie drugiej liczby B
4 B=input('Podaj B: ');
5 //wykonanie sumowania i przypisanie wyniku do zmiennej s
6 s=A+B;
7 //wyświetlenie wartości s na ekranie
8 disp(s);
```

Rys05 – Edytor programu SCILAB

Aby **zapisać** program należy w oknie z Rys05 wybrać opcję: **Plik->Zapisz jako**.

Aby program **uruchomić** należy wybrać z okna z Rys05 opcję: **Wykonaj -> Zapisz i wykonaj**

Jeśli wszystko jest ok. program zostanie uruchomiony w oknie głównym SCILAB'a.

Jeśli pojawią się błędy wówczas użytkownik zostanie o nich powiadomiony.

Na Rys06 przedstawiono wynik uruchomienia programu z Rys05.

```

Scilab 5.5.1 Console

-->exec('H:\Iwona\Dydaktyka\Cz\s_letni_2014_2015\Metody numeryczne
Podaj A:7
Podaj B:8

15.

```

Rys06 – Efekt uruchomienia programu z Rys05.

abs	wartość bezwzględna, moduł
exp	eksponent
log	logarytm naturalny
log10	logarytm o podstawie 10
cos	cosinus (argument w radianach)
sin	sinus (argument w radianach)
sinc	$\frac{\sin(x)}{x}$
tan	tangente (argument w radianach)
cotg	cotangente (argument w radianach)
acos	arccos
asin	arcsin
atan	arctg
cosh	cosinus hiperboliczny
sinh	sinus hiperboliczny
tanh	tangens hiperboliczny
acosh	argch
asinh	argsh
atanh	argth
sqrt	pierwiastek kwadratowy
floor	$E(x) = (\lfloor x \rfloor) = n \Leftrightarrow n \leq x < n + 1, \quad x \in N$
ceil	$\lceil x \rceil = n \Leftrightarrow n - 1 < x \leq n, \quad x \in N$
int	$int(x) = \lfloor x \rfloor$ jeśli $x > 0$ oraz $\lceil x \rceil$ dla $x \leq 0$
erf	funkcja błędu $erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$
erfc	dopełnienie funkcji błędu określone przez $erfc(x) = 1 - erf(x) = \frac{2}{\sqrt{\pi}} \int_x^{+\infty} e^{-t^2} dt$
gamma	$\Gamma(x) = \int_0^{+\infty} t^{x-1} e^{-t} dt$
lngamma	$\ln(\Gamma(x))$
dlgamma	$\frac{d}{dx} \ln(\Gamma(x))$

Rys07 Wybrane funkcje używane w programie Scilab

Zadanie 1

- a) Rozwiąż podane równanie kwadratowe (pisemnie).
- b) Napisz program pobierający od użytkownika współczynniki równania kwadratowego, następnie obliczający pierwiastki podanego równania i wyświetlający je na ekranie.

```
Podaj a:3
Podaj b:1
Podaj c:-5

      1.1350416
- 1.4683749
```

Rys08 – Efekt uruchomienia programu z 1b z **przykładowymi** danymi.**Funkcje**

W przeciwieństwie do skryptów funkcje mogą przyjmować parametry i zwracać wyniki. Również zapisywane są w plikach. W jednym pliku można umieścić wiele funkcji, każda z funkcji musi zaczynać się słowem kluczowym **function**, kończyć **endfunction**.

Wbudowane i dostępne są właściwie wszystkie funkcje elementarne (są już skompilowane). Funkcje trygonometryczne — proste, odwrotne, hiperboliczne... (argumenty tych funkcji są w radianach). Funkcje *logarytmiczne*: log — logarytm naturalny, log10 — logarytm dziesiętny, log2 — logarytm o podstawie 2.

Funkcja *eksponencjalna*: exp.

Liczby *losowe*: funkcja rand(n1,n2) — generuje macierz n1×n2 składającą się z liczb losowych z przedziału <0,1).

Istnieje też *generator liczb losowych* grand umożliwiający uzyskanie różnych zadanych rozkładów.

Funkcje *specjalne*: erfc, funkcja gamma, funkcje Bessela pierwszego i drugiego rodzaju.

Definiowanie funkcji w pliku :

Przykład 2

```
moja_funkcja.sci
1 function [sinus, cosinus]=moja_funkcja(x)
2 sinus=sin(x);
3 cosinus=cos(x);
4 endfunction
```

W tym przypadku funkcja przyjmuje pojedynczy parametr, a zwraca wierszowy wektor dwuelementowy (jeśli argumentem będzie wektor funkcja zwróci macierz, składającą się z dwóch wektorów wynikowych funkcji sin i cos).

```
Scilab 5.5.1 Console
-->[sinus,cosinus]=moja_funkcja(0)
cosinus =

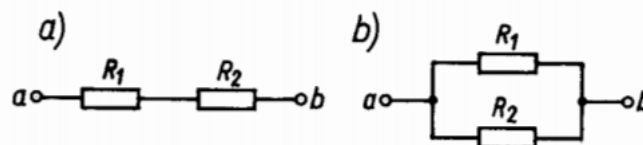
    1.
sinus =

    0.
```

Zadanie 2

- a) Rozwiąż podane zadanie (pisemnie).

1.1. Rezystancja wypadkowa dwóch rezystancji połączonych szeregowo (rys. 1.1a) jest równa $25\ \Omega$, a połączonych równolegle (rys. 1.1b) $4\ \Omega$. Obliczyć wartości poszczególnych rezystancji.



Rys. 1.1

- b) Napisz program (**zdefiniuj funkcję**) obliczający wartości poszczególnych rezystancji (**przykład powyżej**).

Definiowanie funkcji w linii komend:

Przykład 3

```
deff(' [sinus,cosinus]=moja_funkcja(x) ','sinus=sin(x);cosinus=cos(x) ');  
[sinus,cosinus]=moja_funkcja(0)
```

```
-->[sinus,cosinus]  
ans  =  
  
0.    1.
```

Tak zdefiniowana funkcja działa identycznie jak ta zdefiniowana w pliku. Można używać jej również jako argumentu dla innych funkcji.

Ogólnie składnia tego polecenia/definicji jest następująca:

deff(łańcuch_składni,wektor_łańcuchów_poleceń) (ten ostatni wektor może być wierszowy lub kolumnowy).

Zadanie 3

1. Napisz program z wykorzystaniem polecenia `deff` obliczający wartości poszczególnych rezystancji (**przykład powyżej**).

Wynikiem tego ćwiczenia są:

1. Pisemne rozwiązanie podanego równania kwadratowego
2. Program pobierający od użytkownika współczynniki równania kwadratowego, następnie obliczający pierwiastki podanego równania i wyświetlający je na ekranie. **c01_z1_nazwisko**
3. Pisemne rozwiązanie zadania 1.1
4. Program (**zdefiniuj funkcję**) obliczający wartości poszczególnych rezystancji. **c01_z2_nazwisko**
5. Program z wykorzystaniem polecenia `deff` obliczający wartości poszczególnych rezystancji. **c01_z3_nazwisko**

Na koniec zajęć pliki należy spakować programem WinRar odpowiednio pod nazwą:

Kierunek EiEO nazwa pliku: **O_Met_N_STxx_CW01_nazwisko.rar**

Kierunek AiR nazwa pliku: **A_Met_N_STxx_CW01_nazwisko.rar**

Kierunek Elektrotechnika nazwa pliku: **E_Met_N_STxx_CW01_nazwisko.rar**

Kierunek Telekomunikacja nazwa pliku: **T_Met_N_STxx_CW01_nazwisko.rar**

i przestać na platformę e-learningową <https://e-learning.pcz.pl> w ramach kursu

Metody numeryczne - lab - SI 2021/22NS-Z

korzystając z polecenia „Prześlij sprawozdanie z ćwiczenia nr 01”.