

Piotr Chojnacki:

MATLAB

zajęcia 4:

{Poniżej znajduje się zrzut ekranowy pola gdzie wpisuje się komendy. Nie wszystkie komendy są opisane. Wynik działania każdej z nich widać linijkę niżej.}

Znajdziesz tutaj funkcje i skrypty funkcyjne, które można zastosować w chemii a szczególnie w spektroskopii – przykłady.

wczytanie.m

{Zadaniem funkcji jest wczytanie zbiorów widm. Funkcja sama dodaje rozszerzenie w postaci PRN. Dane z plików wczytywane są jako macierz. Gdzie pierwszą kolumną jest liczba falowa a drugą intensywność.}

```
function macierz=wczytanie(nazwa,pocz,krok,kon)
macierz=[]
for i=pocz:krok:kon
    nazwa1=[nazwa, int2str(i),]
    eval(['load ' nazwa1 '.PRN'])
    dane=eval(nazwa1);
    macierz=[macierz dane(:,2)];
end
```

{przykład wywołania – 20...85 są to kolejne nazwy widm dlatego krok wynosi 5}
wczytanie('NBUT',20,5,80);

macierz =

[]

nazwa1 =

NBUT20

nazwa1 =

NBUT25

nazwa1 =

NBUT30

nazwa1 =

NBUT35

nazwa1 =

NBUT40

nazwa1 =

NBUT45

nazwa1 =

NBUT50

nazwa1 =

NBUT55

nazwa1 =

NBUT60

nazwa1 =

NBUT65

nazwa1 =

NBUT70

nazwa1 =

NBUT75

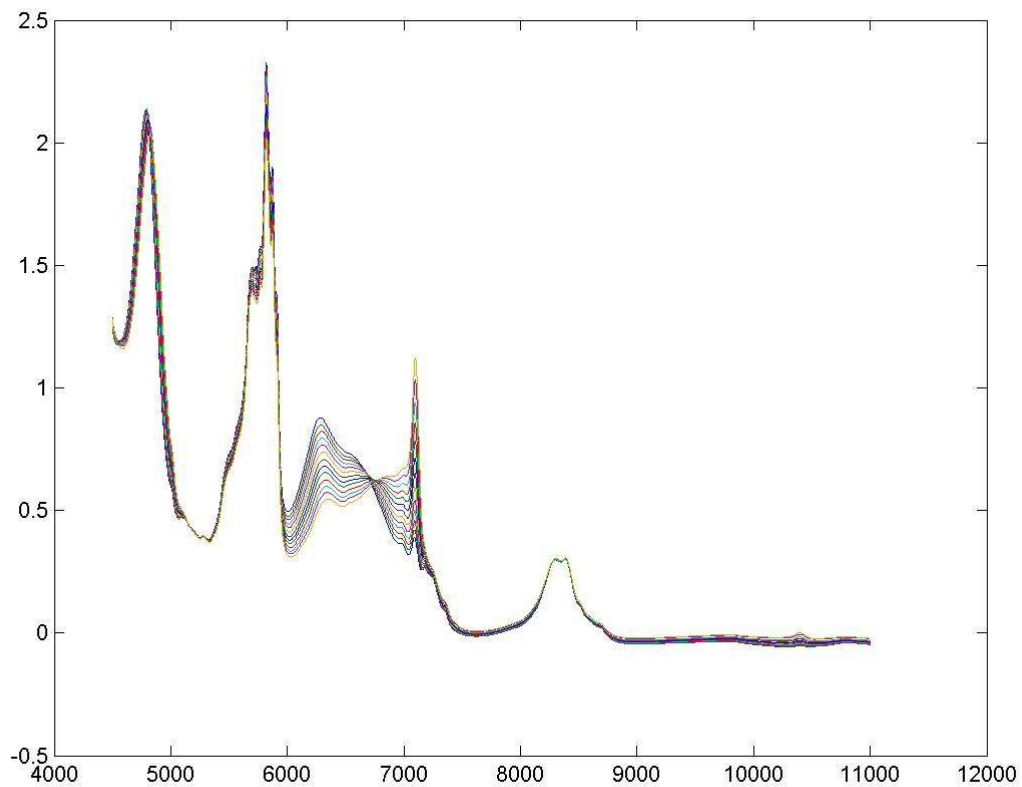
nazwa1 =

wczytanie1.m

{Zadaniem funkcji jest wczytanie zbiorów widm. Funkcja sama dodaje rozszerzenie w postaci PRN. Dane z plików wczytywane są jako macierz. Gdzie pierwszą kolumną jest liczba falowa a drugą intensywność. Dodane zostało rysowanie widma}

```
function [macierz,y]=wczytanie1(nazwa,pocz,krok,kon)
macierz=[]
for i=pocz:krok:kon
    nazwa1=[nazwa, int2str(i),]
    eval(['load ' nazwa1 '.PRN'])
    dane=eval(nazwa1);
    macierz=[macierz dane(:,2)];
end
y=dane(:,1);
plot(y,macierz)
```

{Wynikiem działania funkcji jest następujący wykres wczytanych widm}



wczytanie1a.m oraz wczytanie1b.m

{Zadaniem obu funkcji jest wczytanie zbiorów widm. Funkcje sama dodają rozszerzenie w postaci PRN. Dane z plików wczytywane są jako macierz. Gdzie pierwszą kolumną jest liczba falowa a drugą intensywność. Poza tym aby wczytać i narysować widma oba zbiory muszą mieć tą samą długość.}

```
function [macierz,x]=wczytanie1a(nazwa,pocz,krok,kon)
```

```
macierz=[]
for i=pocz:krok:kon
    nazwa1=[nazwa, int2str(i),]
    eval(['load ' nazwa1 '.PRN'])
    dane=eval(nazwa1);
    [c,y]=size(dane)
    if c>y
        macierz=[macierz dane(:,2)];
    else macierz=[macierz;dane(2,:)];
end
end
if c>y
    x=dane(:,1);
else x=dane(1,:);
end
plot(x,macierz)
```

```
-----
function [macierz,x]=wczytanie1b(nazwa,pocz,krok,kon)
```

```
macierz=[]

w=[nazwa, int2str(pocz),]
eval(['load ' w '.PRN'])
pio=eval(w);

pocz=pocz+krok;
for i=pocz:krok:kon
    nazwa1=[nazwa,int2str(i),]
    eval(['load ' nazwa1 '.PRN'])
    dane=eval(nazwa1);

    if length(dane(:,1))==length(pio(:,1))
        [c,y]=size(dane)
        if c>y
            macierz=[macierz;dane(:,2)];
        else macierz=[macierz;dane(2,:)];
        end
    end
end
```

```

    if c>y
        x=dane(:,1);
        else x=dane(1,:);
    end

    else
        disp('zbiory o różnej długości')
    end
end
end
plot(x,macierz)

```

funnumer.m oraz **numer.m**

{Zadaniem obu funkcji jest podanie pozycji i wartości punktu o danej liczbie falowej. Do prawidłowego działania potrzebna jest funkcja numer.m, z której korzysta funkcja funnumer.m.}

```

function wd=funnumer(macierz)

[macierz,x] = wczytanie1('NBUT', 20,5,20)

lf=input('Podaj liczbę falową [lf] ')
po=numer(x,lf);

figure
plot(x,macierz);
grid on

```

```

-----

function pos=numer(x,lf)

zp=length(lf);
if zp==1
xp=abs(x-lf);
[pom pos]=min(xp);
else
for i=1:zp
xp=abs(x-lf(i));
[pom pos(i)]=min(xp);
end
end
end

```