

```
%*****
% funkce provadejici serazeni vektoru podle velikosti prvku
% a vykresleni s animaci v bar grafech
%*****
% vstup
% radkovy vektor hodnot x
% vystup
% dva parametry
% 1p - serazeny vektor
% 2p - vektor dvou hodnot s minimem a maximem vstupniho vektoru
% volani
% [1p,2p]=michal(x);
% poznamky
% pouziva jen lokalni promenne
%*****
function [V_output,minmax]=michal(V_input)
figure(1) % nachystani si prvni graf
V_output=V_input; %kopie prijateho vektoru do vystupniho
n=length(V_output); %zjisteni delky
for index1=1:n %pruchod od prvnioho do koncoveho prvku
    for index2=index1+1:n %pruchod posunutý o +1 do konce
        if (V_output(index2) < V_output(index1)) %zjisteni který prvek je větší
            pomocna = V_output(index1); %ulozime si prvek index 1 před přehozením
            V_output(index1) = V_output(index2); %prekopirujeme prvek index 2 do 1
            V_output(index2) = pomocna; %prekopirujeme prvek 1 na pozici prvek 2
            pause(0.5); %animujeme 0.5s, pro rychlost zakomentováno
            bar(V_output); %vykreslime animaci po každém kroku v bar grafu
            %bar graf vykresluje postupně přehazování prvku ve vektoru
        end
    end
end
figure(2) %nachystame si druhy graf
dvojity(:,1)=V_input; %ulozime TRANSPONOVANY vektor na 1 sloupec original
dvojity(:,2)=V_output; %ulozime TRANSPONOVANY vektor na 2 sloupec serazeny
bar(dvojity);%vykreslime bargraf jako dvojsloupce, díky vektoru "dvojity"
legend('puvodni','serazeny','location','best') %legenda

%minimum=min(V_input); %vypočty minima
%maximum=max(V_input); %vypočty maxima
minmax=[min(V_input),max(V_input)]; %uložení do vektoru min a max včetně výpočtu
end
```