Octave: příkazy (if, switch, while, do-until, for, break), proměnné (stálé, globální)

If

- příkaz, který dělá rozhodnutí

- jsou tři základní možnosti, jak může vypadat

- vždy musí začínat příkazem “if“ a končit příkazem “endif“

a) if (podmínka)

udělej toto, pokud je podmínka pravdivá

endif

b) if (podmínka)

udělej toto, pokud je podmínka pravdivá

else

udělej toto, pokud není podmínka pravdivá

endif

c) if (podmínka1)

udělej toto, pokud je podmínka1 pravdivá

elseif (podmínka2)

udělej toto, pokud je podmínka2 pravdivá

else

udělej toto, pokud není ani jedna z podmínek pravdivá

endif

- u této varianty je možné udělat libovolné množství podmínek

- příklady k jednotlivým možnostem

a)

a = 10;

if a == 10

disp("a je rovno deseti")

endif

- !!! u prvního a = 10 je pouze jedno “=“, proměnné a přiřazujeme číslo deset

- !!! u příkazu if jsou mezi “a“ a číslem použité dvě rovná se “==“. To proto, že se ptáme, jestli je a identické s desítkou

- pokud je tedy a rovno deseti, vypíše Octave “a je rovno deseti“

- pokud a není rovno deseti, Octave nevypíše nic

b)

a = 11;

if a == 10

disp("a je rovno deseti")

else

disp("a neni rovno deseti")

endif

- v tomto příkladu Octave vypíše “a je rovno deseti“ pokud je a rovno deseti, jinak vypíše “a není rovno deseti“

c)

a = 11;

if a == 10

disp("a je rovno deseti")

elseif a == 11

disp("a je rovno jedenácti")

else

disp("a neni rovno deseti ani jedenácti")

endif

- v tomto příkladu Octave vypíše “a je rovno deseti“ pokud je a rovno deseti nebo “a je rovno jedenácti“ pokud je a rovno jedenácti nebo “a není rovno deseti ani jedenácti“, pokud a není rovno deseti ani jedenácti

While

- nejjednodušší smyčkový příkaz v Octave

- smyčka se opakuje, dokud není porušena podmínka

while (podmínka)

udělej toto, pokud je podmínka splněna a pokračuj, dokud nebude porušena

endwhile

- příklad

a = 10;

while a <= 15

disp(a)

a = a + 1;

endwhile

- přičítej k “a“ jedničku, dokud nebude porušena podmínka, že “a“ je menší nebo rovno deseti – Octave vytiskne 10, 11, 12, 13, 14, 15

- !!! pozor na nekonečnou smyčku

- příklad nekonečné smyčky

a = 10

while a = 10

disp(a)

endwhile

- Octave bude donekonečna tisknout 10 a Vám nezbyde, než ho vypnout natvrdo přes Správce úloh

Do-Until

- to stejné jako while, pouze podmínka je uvedena na konci

do

udělej toto, pokud je podmínka splněna a pokračuj, dokud nebude porušena

until (podmínka)

- příklad

a = 10;

do

disp(a)

a = a + 1;

until(a == 16)

- přičítej k “a“ jedničku, dokud se “a“ nebude rovnat 16 – Octave vytiskne 10, 11, 12, 13, 14, 15

- zde nemůžeme do podmínky dát je menší nebo rovno, protože deset je vždy menší než šestnáct a Octave by tedy nic neudělal (příkaz by znamenal: dělej, dokud nebude 10 < 16)

For

- počítá iterace pro zadaný interval

- všechny smyčky while lze přepsat na smyčky for

- pokud znáte dopředu počet iterací (opakování) je výhodnější for

for i = x0:krok:xn

udělej toto

endfor

- kde x0 znamená, krok znamená a xn znamená, např. 0:1:5 znamená 0, 1, 2, 3, 4, 5

- může se jít i sestupně, např. 10:-1:0 znamená 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0

- příklad

for i = 0:1:5

disp(i)

endfor

- pro i od nuly do pěti, s krokem jedna, zobrazí i

- Octave tedy tiskne 0, 1, 2, 3, 4, 5

Break

- možnost, jak se dostat ze smyček while a for

- tento příkaz lze použít pouze v těle smyčky

- tam kde chcete smyčku ukončit, napíšete break