BPC-UDP Cvičení 07

# Příklad 1

Napište a odlaďte program v jazyce C, který:

* Přečte ze standardního vstupu zadaný počet čísel typu **double** a uloží je do pole pojmenovaného data. Po každém přečtení čísla zkontrolujte, zda číslo leží v povoleném rozsahu <MIN, MAX> (např. <-4.0, +5.0>). Pokud uživatel zadá číslo mimo povolený rozsah, program vypíše na standardní výstup chybové hlášení: *"Zadane cislo je mimo povoleny rozsah!"* a číslo je ignorováno (neukládá se do pole data a nepočítá se do počtu zadaných čísel).
* Nalezne v poli největší a nejmenší číslo.
* Největší a nejmenší číslo a vytiskne na standardní výstup.

1. Pomocí direktivy #define definujte hodnotu MIN a nastavte ji na -4.0 a hodnotu MAX a nastavte ji na 5.0.
2. Napište definici funkce main(). Ve funkci main():
   1. Definujte pole data pro uložení N čísel typu **double**. Hodnotu N nastavte na **6** pomocí direktivy #define.
   2. Vytiskněte na standardní výstup výzvu uživateli *"Zadej X cisel: "* (za *X* dosaďte skutečnou hodnotu definovanou pomocí N).
   3. Přečtěte ze standardního vstupu N čísel typu **double** a uloží je do pole data. Po přečtení každého čísla program zkontroluje, zda je přečtené číslo v povoleném rozsahu <MIN, MAX> (např. <-3.0, +4.0>). Pokud uživatel zadá číslo mimo povolený rozsah, program vypíše na standardní výstup chybové hlášení: *"Zadane cislo je mimo povoleny rozsah!"*. K postupnému čtení čísel využijte nejprve příkaz while a následně příkaz for.
   4. Vytiskněte elementy (prvky) pole data na standardní výstup.
   5. Definujte proměnné min, max, obě typu **double**.
   6. Nalezněte v poli největší a nejmenší číslo a uložte je do proměnných min, max.
   7. Vytiskněte na standardní výstup hodnoty proměnných min, max s doprovodným textem *"min=X max=Y"*, kde za *X* a *Y* dosadíte hodnoty min resp. max.

# Příklad 2

Napište a odlaďte program v jazyce C, který přečte ze standardního vstupu dvě přirozená čísla, pomocí Eukleidova algoritmu nalezne největší společný dělitel těchto čísel a vytiskne ho na standardní výstup. Vyjděte z příkladu 3, cvičení 6.   
Program se bude skládat z funkce main() a funkce gcd() (zkratka *Greatest common divisor*).

1. Napište deklaraci (úplný funkční prototyp) funkce gcd(). Funkce bude mít parametry aU a aW, oba typu **unsigned**. Návratová hodnota funkce bude typu **unsigned**.
2. Napište definici funkce gcd(). Funkce nalezne největší společný dělitel čísel předaných funkci prostřednictvím parametrů aU a aW. Nalezený největší společný dělitel bude mít návratovou hodnotu funkce.
3. Napište definici funkce main().   
   Ve funkci main():
   1. Definujte proměnné u a w obě typu **unsigned**.
   2. Vypište výzvu uživateli "*Zadej dve cisla:* "
   3. Přečtete ze standardního vstupu dvě čísla typu **unsigned**. Čísla uložte do proměnných u a w. Pokud při čtení došlo k chybě, program ukončete s návratovým kódem 1.
   4. Zavolejte funkci gcd() a vytiskne na standardní výstup největší společný dělitel.

# Příklad 3

Napište a odlaďte program v jazyce C, který přečte ze standardního vstupu přirozené číslo reprezentující rok a vytiskne na standardní výstup, zda je rok přestupný (angl. *Leap year*) nebo není přestupný. Vyjděte z příkladu 2, cvičení 6.  
Program se bude skládat z funkce main() a funkce is\_leap\_year().

1. Napište definici **static inline** funkce is\_leap\_year(), která bude mít parametr aYear typu **unsigned**. Návratová hodnota funkce bude typu **bool**.
2. Napište definici funkce main().  
   Ve funkci main():
   1. Definujte proměnnou year typu **unsigned**.
   2. Vypište výzvu uživateli "*Zadej rok:* ".
   3. Přečtete ze standardního vstupu číslo typu **unsigned**. Číslo uložte do proměnné year. Pokud při čtení došlo k chybě, program ukončete s návratovým kódem 1.
   4. Zavolejte funkci is\_leap\_year() a vytiskněte na standardní výstup „*Rok XXXX je prestupny.*“ nebo „*Rok XXXX neni prestupny.*“.

# Příklad 4

Napište a odlaďte program v jazyce C, který sečte dvě matice.

1. Definuje pomocí dvou direktiv **#define** rozměry matic ROWS (2 řádky) a COLS (3 sloupce).
2. Napište definici funkce main().  
   Ve funkci main():
   1. Definuje matice (dvourozměrná pole) matA, matB, matC o rozměrech ROWS a COLS. Prvky matic budou čísla typu **int**.  
      V definici inicializujte matici matA na hodnoty:   
      = {  
       { 1, 2, 3 },  
       { 4, 5, 6 }  
       };  
      a matici matB na hodnoty:  
      = {  
       { 10, 20, 30 },  
       { 40, 50, 60 },  
       };
   2. Vytiskne na standardní výstup prvky matice matA a následně prvky matice matB.
   3. Vypočte prvky matice matC. Pro matC platí: matC = matA + matB.
   4. Vytiskne na standardní výstup prvky matice matC.

# Příklady pro samostatné procvičení probrané látky

# Příklad 5

Upravte příklad 4 tak, aby místo sčítání matic prováděl násobení matic