

Zápočtová úloha – číslicové třídění

Budou tříděna 16-bitová čísla bez znaménka (typ *unsigned short*). Třídění bude v šestnáctkové soustavě.

16-bitové číslo obsahuje 4 šestnáctkové číslice, rozsah je $0000_{16} \dots FFFF_{16}$. Třídění podle každé číslice bude metodou *Counting Sort*.

A .. vstupní pole s tříděnými čísly

B .. výstupní pole (stejný rozsah jako pole A)

n .. počet tříděných čísel

i .. index číslice

Counting_Sort(A,B,n,i)

```
for i←0 to 15
    C[i] ← 0
for j←0 to n-1
    d ← Digit(A[j],i)
    C[d] ← C[d]+1
for i←1 to 15
    C[i] ← C[i]+C[i-1]
for j←n-1 downto 0
    d ← Digit(A[j],i)
    C[d] ← C[d]-1
    B[C[d]] ← A[j]
```

Funkce *Digit* vrací číslici z daného čísla s daným indexem:

```
unsigned Digit(unsigned short A, int i)
{
    return (A>>(4*i)) & 0xF;
}
```

A .. pole s tříděnými čísly

B .. pracovní pole (stejný rozsah jako pole A)

n .. počet tříděných čísel

Radix_Sort(A,B,n)

```
for i←0 to 2 step 2 // cyklus pro jednotlivé číslice
    Counting_Sort(A,B,n,i)
    Counting_Sort(B,A,n,i+1)
```

Nastavte počet tříděných čísel na 50000:

```
#define POCET 50000
```

Přidělte paměť pro pole A a B :

```
typedef unsigned short word;
word *A = (word *)malloc(POCET*sizeof(word));
word *A = new word[POCET]; // možný zápis v jazyce C++
```

Inicializujte generátor náhodných čísel:

```
#include <time.h>
srand(time(0));
```

Naplňte pole A náhodnými čísly:

```
(word)rand()
```

Seřadíte je popsanou metodou. Následně ověřte, že jsou seřaděná:

```
for i←0 to POCET-2
    if A[i] > A[i+1]
        vypsat zprávu "nesetříděná" a čísla A[i], A[i+1]
        break
```

Vypište prvních 40 čísel a posledních 20 čísel z pole A .

Program lze sestavit v jazycích:

- ♦ C (ANSI standard C89, ISO standard C90)
- ♦ C++ (standard C++14)

Soubor se zdrojovým kódem nebo celý projekt pošlete elektronickou poštou v příloze dopisu. Při zasílání do předmětu dopisu uveďte zkratku předmětu ALM1 a své jméno. Příložený soubor se zdrojovým kódem nebo projektem bude mít jméno sestavené z prefixu, který je uveden v následující tabulce, příjmení a křestního jména (příjmení bude před křestním jménem).

čas cvičení	prefix
9:45-11:15	ALM9
11:30-13:00	ALM11

Příklad: ALM9SovaPetr.c, ALM9SovaPetr.cpp, ALM9SovaPetr.rar.