

# Paměť 2

## Základy programování 2

Mgr. Markéta Trnečková, Ph.D.



Palacký University, Olomouc

## ■ předávání argumentů funkci:

- hodnotou
- odkazem = předáváme jeho adresu

## Example

```
// navratovy typ funkce: adresa typu int
int deleni(int a, int b, int *r)
{
    /* r - zbytek po deleni */
    *r = a%b;
    return a/b;
}

int main()
{
    int x,y;
    /* x = 2, y = 3*/
    x = deleni(13,5,&y);
    return 0;
}
```

- `#include <string.h>`
- `void *memcpy(void *mem1, void *mem2, size_t n);`

## Example

```
/* spatne. kopirujeme pouze adresu */
foo = bar;

/* kopie cyklem */
for( int i = 0; i < m; i = i+1)
    foo[ i ] = bar[ i ];

/* kopie pomocí memcpy */
memcpy(foo, bar, sizeof(int)*m);
```

# Další funkce s pamětí

- `#include <string.h>`
- `void *memmove(void *mem1, void *mem2, size_t n);`

## Example

```
/* cyklem */
for( int i = 1; i < m; i = i+1)
    foo[ i-1 ] = foo[ i ];

/* pomocí memmove */
memmove( foo , foo+1, sizeof( int )*(m-1));
```

- `#include <string.h>`
- `void *memset(void *mem, int c, size_t n);`

## Example

```
/* cyklem */
for( int i = 0; i < m; i = i+1)
    foo[ i ] = 0;

/* pomocí memset */
memset( foo ,0 , sizeof( int )*m);
```

- 1 Naprogramujte funkci pro spojení dvou textových řetězců. Argumentem funkce jsou dva textové řetězce ke spojení, funkce vrátí nový řetězec (jako ukazatel).
- 2 Naprogramujte funkci, která jako argument bere pole čísel a vrátí pole jejich druhých mocnin (tedy prvek na indexu i vráceného pole bude mít hodnotu druhé mocniny prvku na indexu i vstupního pole).
- 3 Naprogramujte funkci, která bere jako argument 2 proměnné a provede výměnu hodnot těchto dvou proměnných.
- 4 Napište funkci, která pro 2 zadaná čísla vrátí, zda je možné udělat jejich podíl a pokud ano, vrátí i jejich podíl.
- 5 Napište funkci int porovnej, která porovná 2 předané textové řetězce a vrátí -1, pokud je první řetězec menší než druhý, 0, pokud jsou řetězce shodné, nebo 1, pokud je druhý řetězec menší než první. Při práci s textovými řetězci používejte výhradně ukazatele, operátor dereference a pointerovou aritmetiku.

- 6 (Úloha za 5 bodů, pokud řešení odevzdáte přes MS Teams nebo zašlete na marketa.trneckova@gmail.com do dalšího cvičení) Napište funkci s hlavičkou `char *najdi(char *text, char *podretezec, int n)`. Funkce v daném textovém řetězci `text` vyhledá `n`-tý výskyt zadaného podřetězce `podretezec`. Funkce vrací ukazatel na první znak nalezeného podřetězce nebo konstantu `NULL`, pokud podřetězec `podretezec` nebyl nalezen. Pro řetězec "mama mele maso.", podřetězec "ma" a `n = 3` vrátí adresu prvku na indexu 10.