

FPU - porovnávání

ISU cv 12

<http://www.fit.vutbr.cz/~isakin/isu>



Přesuny čísel/konstant

Přesun čísel/konstant

(int, float) EAX → ST0

```
mov eax, 5
push eax           ; do zásobníku jako lokální proměnnou
fild dword [esp] ; načtení celého čísla do ST0
add esp, 4         ; úklid

call ReadFloat      ; získání float (int) čísla
push eax           ; do zásobníku jako lokální proměnnou
fld dword [esp]   ; načtení float čísla do ST0
add esp, 4         ; úklid
```

(float) ST0 → EAX

```
call ReadDouble     ; načtení malé 64b hodnoty
fst dword [ebp-4] ; ulož 32 číslo z ST0 do lokální proměnné
mov eax, [ebp-4] ; zkopíruj hodnotu z lokální proměnné do EAX
call WriteFloat     ; vypiš 32b float hodnoty z EAX
```

Přesun čísel/konstant

(float) const → ST0

```
push __float32__(42.9) ; vloží 32b číslo na vrchol zásobníku  
fld dword[esp]        ; zkopíruje 32b číslo z vrcholu zásobníku  
add esp, 4            ; smazání čísla
```

(double) const → ST0

```
x dq 123456789.9876  
  
push dword [x+4] ; dolní část čísla  
push dword [x]   ; horní část čísla  
fld qword [esp]  ; load z vrcholu zásobníku  
fstp qword [esp] ; uložení 64bit čísla do zásobníku
```

Ukázka

- cv12_A.asm - předávání hodnot

Úkol

- cv12_1.asm - funkce pro objem kuželu



Porovnání

Porovnání desetinných čísel

```
FCOM                ; porovná st0 s st1
FCOMP               ; porovná st0 s st1 & POP
FCOMPP              ; porovná st0 s st1 & 2x POP
FCOM dword [mem32] ; porovná st0 s [mem32]
FCOMP dword [mem32] ; porovná st0 s [mem32] & POP
FCOM stX            ; porovná st0 s stX
FCOMP stX           ; porovná st0 s stX & POP
```

FTST (s nulou), FXAM (ověření hodnoty), FICOM (int), FICOMP (int+pop)

- porovnání nastaví příznaky C3, C2 a C0 (v FPU)!

Porovnání desetinných čísel

```
FTST      ; porovná st0 s nulou a nastaví příznaky
FSTSW AX  ; uloží obsah stavového registru FPU do registru AX
SAHF      ; uloží AH do registru příznaku (flags)
           ; -> mapování příznaku FPU do FLAGS registru:
           ;   C3 => ZF
           ;   C2 => PF
           ;   C0 => CF
```

Comparison Results	ZF	PF	CF
$ST0 > ST(i)$	0	0	0
$ST0 < ST(i)$	0	0	1
$ST0 = ST(i)$	1	0	0
Unordered	1	1	1

Porovnání desetinných čísel

`FCOMI` ; porovná `st0` s `st1` a překopíruje příznaky do `EFLAGS`

`ja` above ; `ST0 > ST1`

`jb` below ; `ST0 < ST1`

`je` equal ; `ST0 == ST1`

`jp` problem ; error, např. `NaN > 0`

Ukázky

- cv12_B.asm - porovnání a jump
- cv12_C.asm - druhy porovnání

Úkoly

- cv12_2.asm - největší číslo z pole
- cv12_3.asm - největší číslo z 64bit pole
- cv12_4.asm - součet čísel menší než
- cv12_5.asm - sort pole