

Zápočtové úkoly I

13. dubna 2022

V externím assembleru NASM naprogramujte následující funkce nebo programy.

1 Převod textové reprezentace na celočíselnou hodnotu

Naprogramujte funkci `str_to_uint`, která převede standardní řetězec jazyka C na celočíselnou hodnotu. Řetězec je předán jako první argument funkce a jako druhý argument je předán základ číselné soustavy (např. desítková, dvojkové soustavě). Pokud se jedná o soustavu vyšší než desítkovou, jsou jako cifry použity znaky A, B atd. Pokud je předán prázdný řetězec nebo řetězec obsahující neplatné znaky, je výsledek nedefinován.

```
unsigned int str_to_uint(char *s, unsigned int base);
```

Příklad použití:

```
str_to_uint("123", 10)    ==> 123
str_to_uint("1001", 10)   ==> 1001
str_to_uint("1001", 2)    ==> 9
str_to_uint("FE", 16)     ==> 254
```

Bodové hodnocení: 2 body

2 Vypsání řetězce

Naprogramujte funkci `print_str`, která s využitím volání jádra operačního systému vypíše na standardní výstup předaný řetězec.

```
void print_str(char *s);
```

Příklad použití:

```
print_str("abc")    ==> abc
print_str("foo")    ==> foo
```

Bodové hodnocení: 2 body

3 Vytvoření trojúhelníku

Vytvořte funkci `triangel`, která do zvoleného bufferu `buf` zapíše standardní řetězec jazyka C obsahující náčrt rovnoramenného pravoúhlého trojúhelníku s odvěsnami o velikosti `size`. Pro vykreslení hran použijte znak `'*'` a pro výplň znak `'.'`. Předpokládejte, že buffer je dostatečně veliký, aby pojmul daný řetězec. Pokud není, je výsledek nedefinován.

```
void triangel(char *buf, unsigned int size);
```

Příklad použití:

```
triangel(buf, 7);
```

```
buf ==>
```

```
*
**
*.*
*..*
*...*
*....*
*****
```

Bodové hodnocení: 3 body

4 Program na vykreslování trojúhelníků

S využitím funkcí z předchozích úkolů vytvořte aplikaci `triangel-app`, která ze standardního vstupu přečte velikost odvěsen rovnoramenného pravoúhlého trojúhelníku a vykreslí jej na standardní výstup. Aplikace by měla být napsán čistě v assembleru a používat přímá volání jádra operačního systému. Pokud je na vstupu neplatná hodnota nebo hodnota větší než 100, je výsledek nedefinován.

Příklad použití:

```
$ echo 7 | ./assignment-01
```

```
*
**
*.*
*..*
*...*
*....*
*****
```

Bodové hodnocení: 3 body

Podmínky hodnocení

- Kód musí být naprogramovaný pro platformu AMD64 a operační systém Linux.
- Kód by měl být řádně a srozumitelně komentován. Pokud nebudou v kódu komentáře, nebude v případě neúplných řešení možné uznat dílčí body.
- Je nutné dodržet předepsané pojmenování a argumenty funkcí.
- Součástí řešení musí být `Makefile`, kterým půjdou řešení přeložit.
- Úkoly odevzdávejte do 21. dubna 23:59 CEST na email petr.krajca (zavináč) upol.cz s předmětem OS1:zapocet01.
- Úkoly odevzdávejte zabalené jako jeden archiv typu zip nebo tar+gz. Název soubor by měl být ve tvaru: <prijmeni><prvni-znak-jmena>.[zip|tgz].
- Pokud budou odhalena dvě shodná řešení, nebudou body uděleny žádnému z takových řešení.