

# TDIU16: Process- och operativsystemprogrammering

## Argumenthantering i C-program

Klas Arvidsson, Daniel Thorén, Filip Strömbäck

## 1 Mål

Målet med uppgiften är att öva på array- och parameterhantering, samt att bättre se hur detta fungerar i UNIX-system, så att det är enklare att implementera något liknande inuti Pintos senare.

Utöver det ger uppgiften övning på formatsträngar till `printf`, vilket är mycket användbart vid felsökning i Pintos senare.

Se ”Bra att känna till om C” i Pintos-Wiki för mer information.

### 1.1 Uppgift

Skriv ett program som går igenom alla argument som gavs till `main` på kommandoraden. Skriv ut varje argument (ord) i en välformaterad lista. Först på varje rad skriver du ut argumentet vänsterjusterat och sedan skriver du ut längden (antalet tecken) på argumentet högerjusterat. Du får anta att alla argument är som längst 20 tecken. Slutligen skall du skriva ut totala antalet tecken (som ett heltal) och genomsnittliga antalet tecken per argument (som ett flyttal) med två decimalers noggrannhet. Programmet skall returnera totala antalet tecken. Ditt program skall överensstämma med testkörningarna nedan.

Du kan lägga ditt program i exempelvis `standalone/lab01`.

## 2 Körexempel

Terminalkommandon här i kursiv stil för tydlighets skull:

```
$ ./a.out testing one two three
```

```
./a.out          7
testing          7
one              3
two              3
three            5
Total length     25
Average length   5.00
```

```
$ echo $?
```

```
25
```

```
$ ./a.out a b c d
```

```
./a.out          7
a                1
b                1
c                1
d                1
Total length     11
Average length   2.20
```

```
$ echo $?
```

```
11
```

```
$ ./a.out and a sixteencharacter command line argument
```

```
./a.out          7
and              3
a                1
```

sixteencharacter	16
command	7
line	4
argument	8
Total length	46
Average length	6.57

\$ *echo* \$?

46