

Lista 2 (Lab) Termin wysłania na SVN do 29.10.2017

1. (5pt) Wytlumacz jakie pliki zawierają katalogi `/dev` oraz `/proc`. Wykorzystując polecenie `dd` odczytaj pierwszy sektor z dysku głównego (uwaga na prawa dostępu) lub podpiętego pendrive'a i wyświetl przez `hexdump -C`. Z katalogu `proc` wyświetl informacje o pamięci oraz procesorze.
2. (5pt) Zapoznaj się z programem `ps` (`man ps`). Naucz się posługiwać tym programem, aby umieć sprawdzać co najmniej istnienie i parametry wybranych procesów (PID procesu i rodzica, stan procesu, priorytet, nice, ilość pamięci, zużycie czasu procesora). Uruchom też kilka terminali pokaż jakie urządzenie tty wykorzystują. Wykonując komendę `ps axjf` pokaż wszystkie procesy które podpięte są do tych terminali (kolumna TTY).
3. (5pt) Zapoznaj się z kompilatorem języka C (polecenie `gcc`) oraz języka C++ (polecenie `g++`). Uruchom poniższy program w języku C.

```
$ cat > test.c
#include <stdio.h>

int main(int argc, char *argv[])
{
    printf("Hello, World!\n");
    return 0;
}
^D
$ gcc -Wall -pedantic test.c
$ ./a.out
```

Wytlumacz każdy z powyższych kroków. Co oznaczają opcje `-Wall` oraz `-pedantic`? Zobacz `man gcc`.

4. (5pt) Uruchom potok co najmniej trzech poleceń działających przez jakiś zauważalny czas. Mogą to być odpowiednio dobrane polecenia systemowe. Sprawdź poleceniem `ps` i odpowiedz jakie zachodzi pokrewieństwo między tymi procesami (jeśli w ogóle zachodzi)
5. (5pt) Poleceniem `mkfifo` (`man mkfifo`) utwórz potok nazwany (ang. *named FIFO*). Wywołując polecenie `cat` w różnych terminalach spowoduj wpisywanie danych do potoku przez jeden(ne) proces(y), i odczytywanie i wyświetlanie ich przez inne. Zaobserwuj, kiedy polecenie `cat` czytające z potoku czeka na więcej danych, a kiedy

kończy pracę. Analogicznie, kiedy czeka na więcej danych (z klawiatury) polecenie **cat** piszące do potoku?

6. (8pt) Napisz program w języku C, który wykorzystując sekwencje Esc (ang. escape sequences) standardu ANSI wyświetli na ekranie napis "Hello, World!", po kolei we wszystkich dostępnych przez terminal kolorach. Czy terminal może wyświetlić 256 kolorów?
7. (10pt) Napisz potok poleceń, który zamienia wszystkie nazwy plików w danym katalogu (bez podkatalogów) na małe litery, czyli wszystkie duże litery występujące w nazwach plików zostaną zamienione na małe, a małe litery pozostają oczywiście dalej małe.

Przykład konwersji stron podręcznika **man** do formatu PDF:

```
$ man -t ps | ps2pdf - ps.pdf
$ man -t gcc | ps2pdf - gcc.pdf
$ # Dla czcionek 'Courier'
$ zcat $(man -w gcc) | groff -Tps -fC -mandoc | ps2pdf - "gcc.pdf"
```

W przypadku braku programu **ps2pdf**, należy zainstalować pakiet **ghostscript** np. pod archlinuksem:

```
# pacman -S ghostscript
```