

## Podstawowy warsztat informatyka — lista 4

Rozwiązane zadania należy deklарować na kuponie dostępnym na mojej półce i na stronie przedmiotu.

**Zadanie 1.** (1 punkt) Wykonaj w terminalu polecenia:

```
sleep 3600 &
nohup sleep 3600 &
sleep 3600
```

Następnie w drugim terminalu wykonaj polecenie `ps -eF | grep ^user`, wpisując zamiast `user` swoją nazwę użytkownika. Co oznacza wynik tego polecenia?

W pierwszym terminalu wciśnij `ctrl+z`, aby zatrzymać proces, a następnie uruchom polecenie `jobs`. Wznów zatrzymane polecenie komendą `bg`, a następnie `fg`. Zabij to polecenie skrótem `ctrl+c`. Sprawdź powtarzając polecenie w drugim terminalu, czy rzeczywiście umarło.

Następnie zabij dwa pozostałe procesy w pierwszym terminalu — jeden używając polecenia `kill`, drugi `killall`.

**Zadanie 2.** (2 punkty) Zapoznaj się z poleceniem `screen`. Dowiedz się, jak za pomocą tego programu uruchomić polecenie (np. `sleep 180 ; echo Ok`), które będzie działać nawet po zamknięciu okna terminala, w którym było uruchomione. Dowiedz się też, jak później odnaleźć uruchomiony proces (tzn. jak przywrócić *sesję* programu `screen`).

Uruchom jakąś aplikację okienkową w terminalu (np. `xcalc`, `gedit`). Zamknij terminal. Co się stało? Co się zmieni, gdy dodasz przed poleceniem `nohup`? A co się stanie, gdy uruchomisz tę aplikację w `screenie`?

**Zadanie 3.** (2 punkty) Dowiedz się, jak działa polecenie `chmod`, pozwalające na zmiany uprawnień plików. Spraw, by wynikiem polecenia `ls -al | tail -n 3` było (nazwy użytkownika, kolejność plików i język nie muszą się zgadzać, ale daty i rozmiary plików powinny być odpowiednio ustawione):

1. (1 punkt)

```
-rw-r--r-- 1 jmi jmi 6 Oct 11 15:29 .txt
-rw-r--r-- 1 jmi jmi 5 Oct 11 15:29 test.txt
-rw-r--r-- 1 jmi jmi 6 Oct 11 15:29 test2.txt
```

2. (1 punkt)

```
-rwx--x--x 1 jmi jmi 6 Oct 11 15:29 .txt
-----rwx 1 jmi jmi 5 Oct 11 15:29 test.txt
----- 1 jmi jmi 6 Oct 11 15:29 test2.txt
```

**Zadanie 4.** (1 punkt) Polecenie `ping 8.8.8.8` mówi m.in., jak szybko potrafimy połączyć się z serwerem 8.8.8.8. Użyj potoku z poleceniem `grep`, by wypisywać tylko co 10 wynik

```
64 bytes from 8.8.8.8: icmp_seq=10 ttl=116 time=22.9 ms
64 bytes from 8.8.8.8: icmp_seq=20 ttl=116 time=38.8 ms
64 bytes from 8.8.8.8: icmp_seq=30 ttl=116 time=22.8 ms
64 bytes from 8.8.8.8: icmp_seq=40 ttl=116 time=22.8 ms
64 bytes from 8.8.8.8: icmp_seq=50 ttl=116 time=22.9 ms
```

Zwróć uwagę, że polecenia `ping` i `grep` działają równocześnie.

**Zadanie 5.** (1 punkt) Znajdź w katalogu `/usr/share` (i podkatalogach) wszystkie pliki, które zawierają słowo `good` i których nazwa kończy się na `right`. Wyświetl nazwy tych plików. Następnie zapisz jedną sekwencją poleceń oddzielonych znakiem `|` zawartość wszystkich tych plików do pliku `wszystkie`. *Wskazówka:* Wygodnie to zrobić poleceniem `xargs`. Gdy zapytasz o rozwiązanie czata GPT, zapewne zaproponuje Ci on rozwiązanie używające polecenia `find`, które jest dość skomplikowane. Jeśli tak, to zachęć go do uproszczenia polecenia.