

<b>Temat zajęć</b>	Programowanie skryptów powłoki
<b>Zakres materiału</b>	Tworzenie i uruchamianie skryptów powłoki <code>bash</code>

## Materiał teoretyczny

- język skryptowy powłoki `bash`, zmienne środowiskowe, filtry i potoki, programy narzędziowe

## Treść zadania

Program narzędziowy `lsblk` wyświetla zbiorczo informacje o dostępnych w systemie urządzeniach blokowych. Twoim zadaniem będzie napisanie skryptu powłoki `bash` implementującego uproszczoną wersję tego polecenia.

Urządzenie **blokowe** to urządzenie, które pozwala na odczyt/zapis danych wyłącznie porcjami o określonym rozmiarze, czym różni się od urządzenia **znakowego**, operującego na pojedynczych bajtach. Przykładem urządzenia blokowego jest dysk, a znakowego terminal.

Katalog `/sys/block` w systemie Linux jest częścią wirtualnego systemu plików `sysfs`, który zapewnia wygodny interfejs dostępu do informacji o urządzeniach blokowych i ich stanie. Znajdziesz tu informacje o takich dostępnych w systemie urządzeniach jak m.in. dyski twarde, dyski SSD, napędy optyczne, urządzenia pamięci masowej dostępne przez USB oraz wirtualne urządzenia blokowe (tzw. urządzenia `loop`).

Każdy z katalogów istniejących w katalogu `/sys/block` opisuje jedno urządzenie blokowe dostępne aktualnie w systemie, przy czym dla potrzeb niniejszego zadania interesujące są wyłącznie urządzenia dyskowe. Rozpoznasz je po tym, że w swoim katalogu będą mieć plik o nazwie `uevent` zawierający linię tekstu postaci:

```
DEVTYPE=disk
```

Wewnątrz katalogu urządzenia mogą znajdować się podkatalogi opisujące partycje umieszczone na tym urządzeniu. Rozpoznasz je po tym, że ich nazwy są rozwinięciem nazwy urządzenia, np. jeśli katalog urządzenia nazywa się `/sys/block/sda`, to podkatalog partycji na tym urządzeniu może się nazywać np. tak: `/sys/block/sda/sda1`. Możliwy jest także scenariusz, w którym urządzenie blokowe nie jest dzielone na partycje (np. w przypadku urządzeń optycznych) i w takim wypadku podkatalogi partycji nie są tworzone.

W katalogu opisującym partycję urządzenia blokowego interesujące są dla ciebie następujące katalogi i pliki:

- **device** – podkatalog w katalogu `/sys/block` zawierający informację opisującą fizyczną strukturę urządzenia/partycji; brak tego katalogu jest wskazówką, że mamy do czynienia z urządzeniem wirtualnym, a nie fizycznym;

- **uevent** – w linii rozpoczynającej się od prefiksu „DEVTYPE=” powinno znaleźć się słowo „partition”;
- **size** – w pliku znajdziesz liczbę obrazującą rozmiar urządzenia liczony w blokach o rozmiarze 512 bajtów;
- **removable** - w pliku znajdziesz liczbę informującą, czy urządzenie jest wymienne (1) czy też nie (0); brak pliku jest wskazówką, że urządzenie jest niewymienne.

Dysponując powyższymi informacjami napisz skrypt, którego działanie będzie następujące:

- bez opcji skrypt przedstawia się i podaje informację o poprawnym sposobie uruchomienia, np. w następujący sposób:

```
$ ./07.sh
Usage: ./07.sh [-d | -l]
Where:
-d - show block devices and partitions only
-l - show detailed information
```

- z opcją **-d** skrypt wyprowadza nazwy wszystkich fizycznych dyskowych urządzeń blokowych dostępnych w systemie i umieszczonych na nich partycjach, np. w następujący sposób:

```
$ ./07.sh -d

sda
+--sda1
+--sda2
+--sda3

sr0
```

- z opcją **-l** skrypt wyprowadza dodatkowo informację o rozmiarze urządzenia/partycji wyrażoną w mebibajtach oraz o tym, czy urządzenie jest wymienne, np. w następujący sposób:

```
$ ./07.sh -l

sda                153600 MiB fixed
+--sda1            1 MiB fixed
+--sda2            2048 MiB fixed
+--sda3            151549 MiB fixed

sr0                1023 MiB removable
```

- skrypt nie wyświetla żadnych innych informacji i w szczególności ukrywa wszystkie ewentualne komunikaty o błędach zgłaszane przez system operacyjny

- skrypt powinien akceptować tylko jedną z podanych opcji i wyprowadzać komunikat o błędzie, jeśli został uruchomiony z nieprawidłową opcją.

**Uwaga!** Kod źródłowy skryptu (1 plik) po zaprezentowaniu go prowadzącemu zajęcia laboratoryjne musi zostać jako **załącznik** przesłany na adres `sos1@wi.zut.edu.pl`:

- plik z kodem źródłowym musi mieć nazwę: `numer_indeksu.so.lab07.sh` (np. `66666.so.lab07.sh`),
- plik musi zostać wysłany z poczty uczelnianej (domena `zut.edu.pl`),
- temat maila musi mieć postać: `SO IS1 999X LAB07`
  - gdzie `999X` to numer grupy laboratoryjnej (np. `SO IS1 210C LAB07`),
    - w pierwszych czterech liniach skryptu w komentarzach muszą znaleźć się:
      - `shebang`
      - informacja identyczna z zamieszczoną w temacie maila,
      - imię i nazwisko osoby wysyłającej maila,
      - adres e-mail, z którego wysłano wiadomość

np.:

```
#!/bin/bash
# SO IS1 210C LAB07
# Jan Nowak
# nj66666@zut.edu.pl
```

- e-mail nie może zawierać żadnej treści (tylko załącznik).

Dostarczone kody skryptów będą analizowane pod kątem wykrywania plagiatów. Niewysłanie wiadomości, wysłanie jej w formie niezgodnej z powyższymi wymaganiami lub wysłanie pliku, który nie będzie się poprawnie uruchamiał, będzie traktowane jako brak zadania i będzie skutkowało otrzymaniem oceny niedostatecznej.