# Referát z OS

# Keywords z cvičenia 7

## semafor

----------------

Je abstraktny datovy typ a zaroven aj vseobecny synchronizacny prostriedok. Potrebuje podporu v jadre OS. Jeho zakladne

operacie su init, wait, signal. Wait a Signal su zaroven v kritickej oblasti semaforu. Hodnota semaforu urcuje pocet

operacii wait, ktore sa mozu vykonat bez cakania. Nepouziva obsadzujuce cakanie.

## operacie nad semaforom

-------------------------------

init - nastavi pociatocnu hodnotu semaforu, ktora predstavuje, že kolko funkcii wait sa može vykonať bez čakania

wait - zavola sa pred spustenim procesu, ktory vola wait

semafor obsahuje cele nezaporne cislo a ak to cislo je vacsie ako 0, tak sa toto cislo dekrementuje a wait skonci

a dany proces može vstupiť do KO a vykonavať sa.

Ak je toto cislo rovne nule, tak sa proces uspi a zaradi sa do radu čakajucich procesov

signal - zavola sa po skončeni procesu, ked vystupi z KO

pozrie sa na rad procesov, ak je neprazdny, tak sa vyberie jeden z uspatych procesov (ze ktory, to zalezi od implementacie)

a zobudi ho. Ak je rad čakajucich procesov prazdny, tak inkrementuje hodnotu semaforu

sem\_value: 1

P1 wait // dekrementuje na 0

P2 wait // da sa do queue

P1 work

P1 signal // vybere P2

P2 work

P2 signal // inkrementuje na 1

## invariant semaforu

--------------------------

oznacme nw ako pocet dokoncenych operacii wait, ns - ako pocet dokoncenych operacii signal, c ako pociatocnu hodnotu semaforu,

potom plati nasledovny vztah:

nw(sem) <= ns(sem) + c(sem)

## vseobecny semafor

------------------------

nadobuda hodnoty nezaporne cisla

## binarny semafor

-----------------------------

nadobuda len hodnoty 0 a 1 - znaci sa tiez mutex akoze mutual exclusion, sluzi na riesenie vzajomneho vylucovania

## inicialna hodnota semaforu

-----------------------------

predstavuje počet nejakeho prostriedku, kde ten prostriedok je KO a može ho využivať iba max. jeden proces. Že mam recepčnu

na intrakoch a mam 200 klučov od 200 izieb. Kluč od izby je KO, može ho mať len jeden proces, ale mam 200 takych klučov a to

je ta hodnota semaforu.

## monitor

----------------------------

Vychadza sa z konceptu triedy v zmysle OOP. Je potrebna podpora kompilatora. Monitor je udajovy typ a združuje premenne

a operacie nad nimi. Premenne su privatne, zvonku nepristupne. Operacie su verejne. Procesy vstupuju do monitora volanim

jeho operacii. Operacie sa v čase vylučuju.

# Sed

* Slúži na spracovanie textu, ktorý cez neho prúdi.
* Je silný nástroj na transformáciu a úpravu dát, vhodný na naučenie.

## Princíp činnosti:

1. Prečíta riadok
2. Vykoná príkazy
3. Vypíše riadok

## Syntax

sed OPTIONS... [SCRIPT] [INPUTFILE]...

## Príkazy a Adresy

* p: vypísanie riadku
* =: výpis čísla riadku
* n: prejdi na ďalší riadok
* i: vloženie textu pred riadok
* a: vloženie textu za riadok
* c: vloženie textu namiesto riadka
* d: vymazanie riadku
* s/vzorka/nahrada/[flags]: substitúcia, nahrádzanie

## Úlohy

Vypísať všetky riadky okrem siedmeho až desiateho.

sed '7,10d' list.txt

Zmeniť 'Zuzka' za 'Lucka', len v treťom riadku.

sed '3s/Zuzka/Lucka/g' list.txt

Zmeniť 'Zuzka' za 'Lucka' vo vete 'Zuzka, ja Ťa milujem.' (použiť poradie výskytu).

sed '/Zuzka, ja Ta milujem./s/Zuzka/Lucka/g' list.txt

Zmeniť 'Zuzka' za 'ZUZKA'.

sed 's/Zuzka/ZUZKA/g' list.txt

Zmeniť 'Zuzka' za 'zUZKA'.

sed 's/Zuzka/zUZKA/g' list.txt

Zmeniť 'ZUzka' za 'zuZKA'.

sed 's/ZUzka/zuZKA/g' list.txt

Zmeniť 'mozno' za 'nemozno'. 'Nemozno' nech nie je 'nenemozno'.

sed 's/mozno/nemozno/g; s/nenemozno/nemozno/g' list.txt

Vypísať iba báseň tak, že z nej odstrániť každý nepárny riadok.

sed '1d;n;d' list.txt

Vypísať iba báseň tak, že z nej odstrániť každý párny riadok.

sed 'n;d' list.txt

Prepísať v súbore 'list.txt' 'Zuzka' na 'Lucka'.

sed -i 's/Zuzka/Lucka/g' list.txt

sed 's/Zuzka/Lucka/g' list.txt

Vypísať iba báseň s číslami riadkov, ktoré budú na začiatku každého riadka.

sed '=' list.txt | sed 'N;s/\n/ /'

## Bonus

#!/bin/bash

awk '{print $1}' access.log | sort | uniq > original\_ips.txt

anonym\_ip=100

awk '{print $1 " 123.45.67."anon\_ip++}' original\_ips.txt > mapped\_ips.txt

cp access.log anon.log

while IFS= read -r line; do

original\_ip=$(echo $line | awk '{print $1}')

mapped\_ip=$(echo $line | awk '{print $2}')

sed -i "s/$original\_ip/$mapped\_ip/g" anon.log

done < mapped\_ips.txt

awk '{print $1}' anon.log | sort | uniq -c | sort -nr | head