

Јунски испит по
Оперативни системи
21.6.2022



120

МИН.

1. Да се напише PERL скрипта која како аргументи од командна линија добива 2 датотеки (имиња на датотеки). Доколку има помалку од 2 датотеки, тогаш треба да се испише порака за грешка и да заврши скриптата. Првата датотека содржи парови зборови, по два зборови во секој ред посебно, втората датотека е датотека со случаен текст. Паровите зборови на првата датотека се гранични т.е. првиот збор е почетната граница а вториот збор е крајот. Скриптата треба да ги најде и отпечати сите зборови од втората датотека кои се наоѓаат од првиот граничен збор до вториот и да ги отпечати на екран. Ова треба да се направи за сите парови гранични зборови од првата датотека. Доколку некој од граничните зборови го нема во втората датотека или првиот не е пред вториот, во тој случај не се печати за тој пар гранични зборови ништо.
Пример:
Dat1.txt:
prv vtor
most koj
tri kraj

Dat2.txt
Sekoј prv den sedime na most pokraj reka, I sekoј vtor od nas e koj od koj od elitata. Na kraj site se vratija na rabota

На екран:
prv den sedime na most pokraj reka, I sekoј vtor
most pokraj reka, I sekoј vtor od nas e koj
2. Да се напише Shell скрипта која како аргументи од командна линија добива имиња на два директориуми, притоа доколку се повика без аргументи, скриптата испишува порака за грешка, додека пак, доколку се повика со еден аргумент, вториот се смета дека е тековниот директориум. Скриптата треба да ја земе првата текстуална датотека од двата директориуми (со екстензија .txt) и да ја повика PERL скриптата, со тоа што како прв аргумент ќе се прати датотеката од првиот директориум, додека како втор аргумент датотеката од вториот директориум. Доколку двата директориуми се исти, тогаш вториот директориум се заменува со тековниот директориум. На крај да се отпечати резултатот од извршување на PERL скриптата.
3. Да се напише програма во C програмскиот јазик каде како аргументи (од командна линија) добива најпрво бројка X, а потоа листа од команди (без аргументи). Програмата треба да ги изврши во посебни процеси секоја една од командите. Доколку бројот на команди е помал од 5, тогаш се извршуваат сериски (прво првата, па откако ќе заврши втората се извршува третата и т.н.) притоа првата команда чека X секунди пред да почне да се извршува, останатите не чекаат туку веднаш по претходната почнуваат со извршување. Доколку бројот на команди е 5 или повеќе, тогаш сите команди почнуваат паралелно да се извршуваат, така што секоја од командите чека X секунди пред да се изврши. Без разлика како ќе отпочнат со извршување командите, излезот од секоја команда да се редиректира во датотека со име исто како и бројот на процесот што се извршува командата со наставка .txt (пример: доколку бројот на процесот е 1567, тогаш тој процес излезот го редиректира кон датотека со име "1567.txt"). На крај, главниот процес ја печати содржината од датотеките што се креирани од процесите на екран.
4. Да се напише драјвер за комуникација со уред. Уредот има меморија од 1KB. Податоците во меморијата се заклучени иницијално со 2 клучеви така што меморијата мора да се отклучи пред истата да се користи. Доколку меморијата е заклучена не може да се чита ниту да се запишува во драјверот. За отклучување, во драјверот мора да се запише првин клучот, па доколку е точен, следната операција кон драјверот е овозможена. Првиот клуч има вредност „juni123“, додека пак, вториот клуч има вредност „juni!@#“. Доколку се запише првиот клуч, драјверот е отклучен само за една операција после клучот, и после повторно е заклучен истиот. Доколку се запише вториот клуч, драјверот е отклучен за повеќе операции после тоа, т.е. е отворен за операции додека повторно не се запише вториот клуч во драјверот.

