



1. Да се напише PERL скрипта со име „prva.plx“ која како аргументи добива најпрво збор кој ќе го пребарува, опција (знаменце) и име на датотека во која ќе се прави пребарувањето. Скриптата треба да го пребара зборот (што се праќа како прв аргумент) во зависност од опцијата (што се праќа како втор аргумент) во датотеката (чие име се праќа како трет аргумент). Опцијата може да биде „l“, „w“ или „f“, притоа опцијата „l“ значи дека треба да се вратат броевите од линиите во кои е пронајден зборот, „w“ како резултат враќа колку зборови биле пронајдени пред зборот што се бара, и опцијата „f“ враќа колку пати е пронајден зборот во датотеката. Доколку зборот го нема во датотеката, опцијата „l“ враќа -1, додека пак опциите „w“ и „f“ враќаат соодветно вредност 0. Зборовите се одделени еден од друг со празно место. Доколку добие невалидна опција, скриптата прекинува со извршување.

Пример:

dat.txt:

Also when you use plain Lorem ipsum text, your design will look like a million other designs out there.

With Random Text Generator your designs will look more unique while still containing text which truly means nothing.

perl prvа.plx will l dat.txt

Резултат: 1 2

perl prvа.plx will w dat.txt

Резултат: 10 24

perl prvа.plx will f dat.txt

Резултат: 2

2. Да се напише SHELL скрипта која како аргумент добива име на ПЕРЛ скрипта, збор за пребарување и листа од имиња на датотеки. Доколку не добие како прв аргумент име на ПЕРЛ скрипта (аргумент што завршува на .plx) тогаш зема како име на скрипта „prva.plx“. Shell скриптата треба да провери дали има ПЕРЛ скрипта со такво име во тековниот директориум или поддиректориумите и доколку нема, завршува Shell скриптата со извршување. Доколку постои ПЕРЛ скрипта со такво име (тековниот директориум или поддиректориумите), тогаш Shell скриптата треба со помош на PERL скриптата, да најде и отпечати на екран, колку пати ќе го пронајде зборот (што се праќа како втор аргумент на Shell) за сите датотеки што се праќаат како аргументи (од трет аргумент па натаму). На крај скриптата треба да отпечати колку вкупно бил пронајден зборот во сите датотеки.

Пример:

./prva.sh prvа.plx will dat1.txt dat2.txt dat3.txt

dat1.txt 7

dat2.txt 0

dat3.txt 12

Vkupno: 19

3. Да се напише програма во C која работи со процеси и нитки. Програмата (главната нитка) како аргументи од командна линија добива збор што се пребарува и листа од имиња на датотеки. Доколку не добие листа од датотеки од командна линија (добие

само збор за пребарување), програмата треба да му дозволи да внесе имиња на 3 датотеки од тастатура. Доколку нема ни збор за пребарување, најпрво го внесува зборот од тастатура, а потоа имињата на 3те датотеки. За секоја датотека во која се прави пребарување, се креира посебна нитка која го прави пребарувањето, притоа, на секоја нитка ѝ се испраќа кој е зборот што треба да го пребарува и името на датотеката во која треба да го прави пребарувањето. Секоја нитка го пребарува зборот во својата датотека, и штом заврши, на екран го печати името на датотеката и колку пати се појавил зборот. Откако ќе завршат сите нитки, главната нитка на екран печати, за секоја датотека одделно, колку процентуално се појавил зборот во таа датотека (процент од вкупното појавување на зборот во сите датотеки).