

Зад. 1. Сменете вашият prompt с нещо по желание. После върнете оригиналния обратно.

----

Зад. 2. Редактирайте вашият `.bash_profile` файл, за да ви поздравява (или да изпълнява някаква команда по ваш избор) всеки път, когато влезете в системата.

----

Зад. 3. Направете си ваш псевдоним (`alias`) на полезна команда.

Да се напише shell скрипт, който:

----

Зад. 4. Приканва потребителя да въведе име (низ) и изпечатва "Hello, името".

----

Зад. 5. Извежда на стандартния изход факултетния номер на потребителя, който изпълнява скрипта, или казва "DANGER! It's a teacher!", ако потребителското име не е факултетен номер.

----

Зад. 6. Приканва потребителя да въведе низ - потребителско име на потребител от системата - след което извежда на стандартния изход колко пъти потребителят се е логнал.

----

Зад. 7. Приканва потребителя да въведе пълното име на директория и извежда на стандартния изход подходящо съобщение за броя на всички файлове и директории в нея.

----

Зад. 8. Чете от стандартния вход имената на 3 файла, обединява редовете на първите два, подрежда ги по азбучен ред и резултата записва в третия файл.

----

Зад. 9. Чете от стандартния вход име на файл и символен низ, проверява дали низа се съдържа във файла и извежда на стандартния изход кода на завършване на командата.

----

Зад. 10. Проверява дали подаденият му параметър се състои само от букви и цифри.

----

Зад. 11. Цветно принтиране. Напишете shell script `color_print`, който взима два параметъра.

Първият може да е измежду "-r", "-g" "-b", а вторият е произволен string.

На командата "echo" може да се подаде код на цвят, който ще оцвети текста в определения цвят.

В зависимост от първия аргумент, изпринтете втория аргумент в определения цвят:

"-r" е червено. Кодът на червеното е '\033[0;31m' (echo -e "\033[0;31m This is red")

"-g" е зелено. Кодът на зеленото е '\033[0;32m' (echo -e "\033[0;32m This is green")

"-b" е синьо. Кодът на синьото е '\033[0;34m' (echo -e "\033[0;34m This is blue")

Ако е подадена друга буква изпишете "Unknown colour", а ако изобщо не е подаден аргумент за цвят, просто изпишете текста.

Hint:

В края на скипта си напишете:

```
echo '\033[0m'
```

,за да не се прецакат цветовете на терминала. Това е цветът на "няма цвят".

----

Зад. 11\*

Малко по-сложна версия на задача 11

Този път програмата ви ще приема само един параметър, който е измежду ("-r", "-b", "-g", "-x").

Напишете shell script, който приема редовете от stdin и ги изпринтва всеки ред с редуващ се цвят. Цветовете вървят RED-GREEN-BLUE и цветът на първия ред се определя от аргумента.

Ако е подаден аргумент "-x", то не трябва да променяте цветовете в терминала (т.е., все едно сте извикали командата cat).

Hint: Не забравяйте да връщате цветовете в терминала.

---

Зад. 12. Напишете shell script guess, която си намисля число, което вие трябва да познате. В зависимост от вашия отговор, програмата трябва да ви казва "надолу" или "нагоре", докато не познате числото. Когато го познаете, програмата да ви казва с колко опита сте успели.

```
./guess (програмата си намисля 5)
```

```
Guess? 22
```

```
...smaller!
```

```
Guess? 1
```

```
...bigger!
```

```
Guess? 4
```

...bigger!

Guess? 6

...smaller!

Guess? 5

RIGHT! Guessed 5 in 5 tries!

Hint: Един начин да направите рандъм число е с  $((\text{RANDOM \% } b) + a)$ , което ще генерира число в интервала  $[a, b]$ . Може да вземете  $a$  и  $b$  като параметри, но не забравяйте да направите проверката.

Да се напише shell скрипт, който:

----

Зад. 13. Получава при стартиране като параметър в командния ред идентификатор на потребител. Скриптът периодично (`sleep(1)`) да проверява дали потребителят е `log-нат`, и ако да - да прекратява изпълнението си, извеждайки на стандартния изход подходящо съобщение.

----

Зад. 14. Приема произволен брой аргументи - имена на файлове. Скриптът да прочита от стандартния вход символен низ и за всеки от зададените файлове извежда по подходящ начин на стандартния изход броя на редовете, които съдържат низа.

----

Зад. 15. Приема два параметъра - име на директория и число. Скриптът да извежда на стандартния изход имената на всички обикновени файлове във директорията, които имат размер, по-голям от подаденото число.

----

Зад. 16. Приема произволен брой аргументи - имена на файлове или директории.

Скриптът да извежда за всеки аргумент подходящо съобщение:

- дали е файл, който може да прочетем

- ако е директория - имената на файловете в нея, които имат размер, по-малък от броя на файловете в директорията.

----

Зад. 17. Приема два параметъра - име на директория и число. Скриптът да извежда сумата от размерите на файловете в директорията, които имат размер, по-голям от подаденото число.

----

Зад. 18. Приема параметър - име на потребител. Скриптът да прекратява изпълнението на всички текущо работещи процеси на дадения потребител, и да извежда колко са били те.