

Име: _____ ФН: _____ Спец.: __ Курс: __ Група: __

Задача	1	2	3	4	Общо
получени точки					
максимум точки	30	30	30	30	120

Забележка: За отлична оценка са достатъчни 100 точки!

Задача 1. Даден е текстов файл с име philip-j-fry.txt. Напишете shell script и/или серия от команди, които извеждат броя редове, съдържащи поне една четна цифра и несъдържащи малка латинска буква от 'a' до 'w'.

Примерно съдържание на файла:

```
123abv123
123zz123
MMU_2.4
```

Примерен изход:

Броят на търсените редове е 2

Задача 2. Файловете във вашата home директория съдържат информация за музикални албуми и имат специфична структура. Началото на всеки ред е годината на издаване на албума, непосредствено следва името на изпълителя на песента. Имената на файловете се състоят от една дума, която съвпада с името на изпълнителя.

Примерно съдържание на файл с име "Bonnie":

```
2005г. Bonnie - "God Was in the Water" (Randall Bramblett, Davis Causey) - 5:17
2005г. Bonnie - "Love on One Condition" (Jon Cleary) - 3:43
2005г. Bonnie - "So Close" (Tony Arata, George Marinelli, Pete Wasner) - 3:22
2005г. Bonnie - "Trinkets" (Emory Joseph) - 5:02
2005г. Bonnie - "Crooked Crown" (David Batteau, Maia Sharp) - 3:49
2005г. Bonnie - "Unnecessarily Mercenary" (Jon Cleary) - 3:51
2005г. Bonnie - "I Will Not Be Broken" - "Deep Water" (John Capek, Marc Jordan) - 3:58
```

Да се състави процедура на bash приемаща два параметъра, които са имена на файлове от вашата home директория. Скриптът сравнява, кой от двата файла има повече на брой редове, съдържащи неговото име (на файла). За файлът победител изпълнете следните действия:

- извлекете съдържанието му, без годината на издаване на албума и без името на изпълнителя
- сортирайте лексикографски извлеченото съдържание и го запишете във файл с име 'изпълнител.songs'

Примерен изходен файл (с име Bonnie.songs):

```
"Crooked Crown" (David Batteau, Maia Sharp) - 3:49
"God Was in the Water" (Randall Bramblett, Davis Causey) - 5:17
"I Will Not Be Broken" - "Deep Water" (John Capek, Marc Jordan) - 3:58
"Love on One Condition" (Jon Cleary) - 3:43
"So Close" (Tony Arata, George Marinelli, Pete Wasner) - 3:22
"Trinkets" (Emory Joseph) - 5:02
"Unnecessarily Mercenary" (Jon Cleary) - 3:51
```

Задача 3. В текущата директория има само обикновени файлове (без директории). Да се напише bash script, който приема 2 позиционни параметъра – числа, които мести файловете от текущата директория към нови директории (a, b и c, които трябва да бъдат създадени), като определен файл се мести към директория 'a', само ако той има по-малко редове от първи позиционен параметър, мести към директория 'b', ако редове са между първи и втори позиционен параметър и в 'c' в останалите случаи.

Задача 4. Всеки от процесите P и Q изпълнява поредица от три инструкции:

```
process P                process Q
p_1                      q_1
p_2                      q_2
p_3                      q_3
```

Осигурете чрез семафори синхронизация на P и Q така, че инструкция p_1 да се изпълни преди q_2, а q_2 да се изпълни преди p_3.

Решения

Задача 1. Пресмятане на броя търсени редове:

```
grep '[02468]' philip-j-fry.txt | grep -v '[a-w]' | wc -l
```

Красив изход с една команда:

```
echo 'Броят на търсените редове е' $(grep '[02468]' philip-j-fry.txt | grep -v '[a-w]' | wc -l)
```

Задача 2. Примерно решение:

```
#!/bin/bash
if [ $# -ne 2 ]
then
    echo "Incorrect number of arguments! "
    exit 99
fi

a=$(grep $1 ~/$1 | wc -l)
b=$(grep $2 ~/$2 | wc -l)

if [ $a -gt $b ]
then
    cut -f4- -d" " ~/$1 | sort > $1.songs
elif [ $a -lt $b ]
then
    cut -f4- -d" " ~/$2 | sort > $2.songs
else
    echo "There is no winner. "
fi
```

Задача 3. Задачата се решава от командна процедура task3.sh:

```
files=$(ls)
mkdir a b c
for i in $files
do
    size=$(cat $i | wc -l)
    if [ $size -lt $1 ]
    then
        mv $i a
    elif [ $size -le $2 ]
    then
        mv $i b
    else
        mv $i c
    fi
done
```

Задача 4. За двата момента на синхронизация използваме два семафора – t1 и t2, инициализираме ги с блокиращо начално състояние:

```
semaphore t1,t2
t1.init(0)
t2.init(0)
```

Добавяме в кода на процесите P и Q синхронизиращи инструкции:

process P	process Q
p_1	q_1
t1.signal()	t1.wait()
p_2	q_2
t2.wait()	t2.signal()
p_3	q_3

Инструкцията `q_2` ще се изпълни след като броячът на семафора `t1` стане положителен. Това се случва след изпълнението на ред `t1.signal()`, който следва инструкция `p_1`.

Аналогично, инструкцията `p_3` ще се изпълни след като броячът на семафора `t2` стане положителен. Това се случва след изпълнението на ред `t2.signal()`, който следва инструкция `q_2`.