

Programming Systems and Environments - Lab 8

Jakub Grzana, 241530

Task 1 - FileDataGenerator

This task defeated me. I've made file data generator: script that generates set of files in given directory with randomly chosen name and surname from files (names passed as argument) and with random values assigned to them. Additionally, one file can store multiple people this way. That being said, I didn't make function to process that data.

Task 2 - NumericDataGenerator

To create 10 digit random number, I've generated two 4-digits random numbers, and one number with 2 digits, then concatenated them like strings. Prime check of such numbers in Bash takes unreasonable amount of time. Bash is language for automation, not for computations so I don't recommend using that program.

Bibliography:

Random numbers in Bash [access 25.05.2022]

<https://tldp.org/LDP/abs/html/randomvar.html>

Prime check in Bash [access 25.05.2022]

<https://www.tutorialsandyou.com/bash-shell-scripting/prime-number-16.html>

Code for task 1:

```
# $1 - number of files to generate
# $2 - number of ppl per file
# $3 - directory
# $4 - file with first names
# $5 - file with second names
```

```
random() {
    divisor=$(expr $2 - $1 + 1)
    val=$(expr $RANDOM % $divisor + $1)
    echo $val
}
```

```
generate() {
    filenum=$1
    pplnum=$2
    dir=$3
    firstnames=$4
    seconnames=$5

    mkdir -p $dir

    for (( i=1; i <= $filenum; i++ ))
    do
        output="File$(printf "%03d" "$i").txt"
        output=$dir/'$output
        for (( j=1; j <= $pplnum; j++ ))
        do
            shuf -n 1 $firstnames >> $output
            shuf -n 1 $seconnames >> $output
            echo $(random 10 100) >> $output
            echo $(random 21 65) >> $output
            echo "" >> $output
        done
        echo Generated file $output
    done
}
```

```
generate $1 $2 $3 $4 $5
```

Code for task 2:

```
# $1 filename
# $2 how many numbers
# $3 output file for processing

random() {
    divisor=$((expr $2 - $1 + 1))
    val=$((expr $RANDOM % $divisor + $1))
    echo $val
}

random10digits() {
    echo "$(random 1000 9999)$(random 1000 9999)$(random 10 99)"
}

generate() {
    filename=$1
    num=$2
    for (( i=1; i <=$num; i++ ))
    do
        echo $(random10digits) >> $filename
    done
}

generate $1 $2

primecheck() {
    num=$1
    for((i=2; i<=num/2; i++))
    do
        if [ $((num%i)) -eq 0 ]
        then
            echo 0
            exit
        fi
    done
    echo 1
}

process() {
    filename=$1
    outname=$2
    while IFS= read -r line
    do
```

```
        output=$(primecheck $line)
        if [ $output -eq "1" ]; then
            echo $line >> $outname
        fi
    done < "$filename"
}

process $1 $3
```