

# Лабораторная работа № 12

Программирование в командном процессоре ОС UNIX. Командные файлы

---

Павличенко Родион Андреевич

Российский университет дружбы народов, Москва, Россия

## Информация

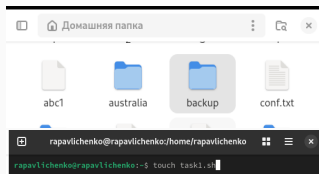
---

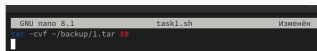
- Павличенко Родион Андреевич
- студент
- Российский университет дружбы народов
- 1132246838@pfur.ru

## Выполнение лабораторной работы

---

## Создаем папку backup и файл task1.txt





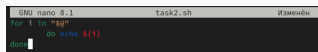
```
GNU nano 8.1 task1.sh Изменён
tar -cvf ~/backup/1.tar $0
█
```

```
rapavlichenko@rapavlichenko:~$ chmod 777 task1.sh
rapavlichenko@rapavlichenko:~$ ./task1.sh
./task1.sh
rapavlichenko@rapavlichenko:~$
```

## Создаем файл task2.txt и выдаем права

```
rapavlichenko@rapavlichenko:~$ touch task2.sh
rapavlichenko@rapavlichenko:~$ chmod 777 task1.sh
rapavlichenko@rapavlichenko:~$ chmod 777 task2.sh
rapavlichenko@rapavlichenko:~$ nano task2.sh
```





The screenshot shows a terminal window with the nano text editor. The title bar at the top reads "GNU nano 8.1 task2.sh Изменён". The editor content consists of a shell script with two lines: "for i in \$(ls)" and "do echo \${i}" on the same line, followed by "done" on the next line. A white cursor is positioned at the end of the "done" line.

```
GNU nano 8.1 task2.sh Изменён
for i in $(ls)
do echo ${i}
done
```

## Создаем файл task3.txt и выдаем права

```
rapaulichenko@rapaulichenko:~$ touch task3.sh
rapaulichenko@rapaulichenko:~$ chmod 777 task3.sh
rapaulichenko@rapaulichenko:~$ nano
```

## Пишем код в файле

```
GNU nano 8.1 task3.sh Изменен
echo "$1/ " | tr -d "\n";
stat --printf "%sA" "$1/";
echo
for i in $(ls)
do echo "$i/ " | tr -d "\n";
  stat --printf "%sA" "$i/";
  echo
done
```

## Создаем файл task4.txt и выдаем права

```
rapavlichenko@rapavlichenko:~$ touch task4.sh  
rapavlichenko@rapavlichenko:~$ chmod 777 task4.sh  
rapavlichenko@rapavlichenko:~$ nano task4.sh
```

## Пишем код в файле

```
CNU nano 8.1                                l.sh
vi/bin/bash

while getopts "ri:oi:pCn" opt; do
  case ${opt} in
    i )
      inputfile=${OPTARG}
      ;;
    o )
      outputfile=${OPTARG}
      ;;
    p )
      pattern=${OPTARG}
      ;;
    C )
      case_sensitive=true
      ;;
    n )
      line_numbers=true
      ;;
    \? )
      echo "Неверный параметр: $opt" 1>&2
      exit 1
      ;;
    : )
      echo "Отсутствует значение для параметра: $opt" 1>&2
      exit 1
      ;;
  esac
done

if [ -z "$pattern" ]; then
  echo "Не указан шаблон для поиска." 1>&2
  exit 1
fi

if [ -z "$inputfile" ]; then
  echo "Не указан входной файл." 1>&2
  exit 1
fi

if [ "$case_sensitive" = true ]; then
  grep_options+=" -i"
fi

if [ "$line_numbers" = true ]; then
  grep_options+=" -n"
fi

grep $grep_options "$pattern" "$inputfile"

if [ ! -z "$outputfile" ]; then
  grep $grep_options "$pattern" "$inputfile" > "$outputfile"
fi
```

Изучили основы программирования в оболочке ОС UNIX/Linux. Научились писать небольшие командные файлы.