

Структура научной презентации

Простейший шаблон

Кулябов Д. С.

01 января 1970

Российский университет дружбы народов, Москва, Россия

Объединённый институт ядерных исследований, Дубна, Россия

Информация

- Мурзаев Замир Зейнадинович
- д.ф.-м.н., профессор
- профессор кафедры прикладной информатики и теории вероятностей
- Российский университет дружбы народов
- kulyabov-ds@rudn.ru
- <https://yamadharma.github.io/ru/>



Цель работы

Цель - изучить основы программирования в оболочке ОС UNIX, научиться писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

Выполнение лабораторной работы

1)Пишем программу, которая анализирует командную строку, а затем ищет нужные строки

```
#!/bin/bash

#Unpacking parameters
while getopts i:o:p:Cn flag
do
    case $flag in
        i) inputFile=$OPTARG;;
        o) outputFile=$OPTARG;;
        p) pattern=$OPTARG;;
        C) C='--color=always'; echo Flag -$flag will switch color output on;;
        n) n=n;;
        *) echo Illegal option $flag used!;;
    esac
done

touch $outputFile
grep $C -${n}e $pattern $inputFile > $outputFile
```

Рис. 1: Командный файл

2)Пишем код на языке С (рис. (fig:002?)) и командный файл, который взаимодействует с ним



The image shows a screenshot of a code editor window. The title bar at the top right indicates the file name is "prog2.c". Below the title bar, on the left, are buttons for "Открыть" (Open) and a dropdown menu. In the center of the title bar is an icon representing a file or folder. Below the title bar, the file path is displayed as "~/work/study/2022-2023/Операционные системы". The main area of the window contains C code with syntax highlighting. The code includes headers for stdio.h and stdlib.h, and defines a main function that prompts the user to enter a number, reads it, and then checks for specific values to exit the program.

```
#include <stdio.h>
#include <stdlib.h>

int main() {
    printf("Введите число: ");
    int a;
    scanf("%d", &a);
    if (a<0) exit(0);
    if (a>0) exit(1);
    if (a==0) exit(2);
    return 0;
}
```


3)Пишем файл, который генерирует определенное количество файлов

```
prog3.sh
File Edit Options Buffers Tools Sh-Script Outline Hide/Show Help
#!/bin/bash

opt=$1;
form=$2;
num=$3;
function Files() {
    for ((i=1; i≤$num; i++)) do
        file=$(echo $form | tr '#' "$i")
        if [ $opt = "-r" ]
        then
            rm -f $file
        elif [ $opt = "-c" ]
        then
            touch $file
        fi
    done
}
Files
```

Рис. 4: Командный файл

4)Пишем файл, который запаковывает в архив все файлы в директории

prog4.sh - GNU

File Edit Options Buffers Tools Sh-Script

```
#!/bin/bash
```

```
files=$(find ./ -maxdepth 1 -mtime -7)
```

```
listing=""
```

```
for file in "$files" ; do
```

```
    file=$(echo "$file" | cut -c 3-)
```

```
    listing="$listing $file"
```

```
done
```

```
dir=$(basename $(pwd))
```

```
tar -cvf $dir.tar $listing
```

Выводы

Изучены основы программирования в оболочке ОС UNIX и приобретены навыки по написанию более сложных командных файлов с использованием логических управляющих конструкций и циклов.

⋮