

**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ СВЯЗИ И МАССОВЫХ
КОММУНИКАЦИЙ**

**Ордена Трудового Красного Знамени
Федеральное государственное бюджетное образовательное учреждение
высшего образования**

«Московский технический университет связи и информатики»

Кафедра «Математическая кибернетика и информационные технологии»

Лабораторная работа №1
по дисциплине
Функциональное программирование
на тему
«Работа с Linux: терминал и администрирование»

Выполнил: студент группы
БВТ2201
Шамсутдинов Рустам

Москва 2025

Введение

Цель работы:

Познакомиться с основными командами операционной системы Linux, освоить работу с терминалом и изучить структуру файловой системы. Необходимо научиться устанавливать и настраивать окружение для разработки на языке Scala, а также приобрести базовые навыки взаимодействия с консольным интерфейсом и пакетным менеджером.

Теоретические основы:

- Основы Linux и командной строки.
- Установка и настройка Scala и sbt.
- Навигация по файловой системе, права доступа, работа с переменными окружения.

Задания:

1. Установить Scala и sbt на свою систему.
2. Создать простое Scala-приложение через sbt и запустить его из терминала.
3. Настроить переменные окружения (PATH и др.).
4. Создать bash-скрипт для автоматического запуска Scala-программы.

Индивидуальное задание:

Вариант 4 – написать скрипт установки Scala и sbt.

Ход работы

Задание 1. Установить Scala и sbt на свою систему.

The screenshot shows the Scala download page for Linux. It includes tabs for macOS, Linux (which is selected), Windows, and Other. Below the tabs, instructions say to run a command in the terminal. For x86_64 architecture, the command is:

```
curl -fL https://github.com/coursier/coursier/releases/latest/download/cs-x86_64-pc-linu
```

For ARM64 architecture, the command is:

```
curl -fL https://github.com/VirtusLab/coursier-m1/releases/latest/download/cs-aarch64-pc
```

A note in a red-bordered box says: "i You may need to restart your terminal, log out, or reboot in order for the changes to take effect."

Below this, a blue bar says "Testing your setup". A message says: "Check your setup with the command `scala -version`, which should output:"

```
$ scala -version
Scala code runner version: 1.4.3
Scala version (default): 3.7.3
```

Рис. 1 – команда для установки.

```
sypo0@pop-os:~/mtuci/mtuci-scala$ curl -fL https://github.com/coursier/coursier/releases/latest/download/cs-x86_64-pc-linux.gz | gzip -d > cs && chmod +x cs && ./cs setup
% Total    % Received % Xferd  Average Speed   Time     Time      Current
          Dload  Upload Total Spent   Left Speed
0       0     0     0       0      0  --::-- --::-- --::-- 0
0       0     0     0       0      0  --::-- --::-- --::-- 0
100 22.2M 100 22.2M 0       0 5921k    0 0:00:03 0:00:03 8330k
Checking if a JVM is installed
https://github.com/coursier/jvm-index/raw/master/index.json
100.0% [#####] 2.8 MiB (2.6 MiB / s)
No JVM found, should we try to install one? [Y/n] Y
Should we update ~/.profile? [Y/n] Y
Some shell configuration files were updated. It is recommended to close this terminal once the setup command is done, and open a new one for the changes to be taken into account.

Checking if ~/local/share/coursier/bin is in PATH
Should we add ~/local/share/coursier/bin to your PATH via ~/.profile? [Y/n] Y

Checking if the standard Scala applications are installed
Installed ammonite
Installed cs
Installed coursier
Installed scala
Installed scalac
Installed scala-cli
Installed sbt
Installed sbt
Installed scalafmt

sypo0@pop-os:~/mtuci/mtuci-scala$
```

Рис. 2 – успешная установка Scala и sbt.

```

sypoo@pop-os:~/mtuci/mtuci-scala$ cat ~/.profile
# ~/.profile: executed by the command interpreter for login shells.
# This file is not read by bash(1), if ~/.bash_profile or ~/.bash_login
# exists.
# see /usr/share/doc/bash/examples/startup-files for examples.
# the files are located in the bash-doc package.

# the default umask is set in /etc/profile; for setting the umask
# for ssh logins, install and configure the libpam-umask package.
#umask 022

# if running bash
if [ -n "$BASH_VERSION" ]; then
    # include .bashrc if it exists
    if [ -f "$HOME/.bashrc" ]; then
        . "$HOME/.bashrc"
    fi
fi

# set PATH so it includes user's private bin if it exists
if [ -d "$HOME/bin" ]; then
    PATH="$HOME/bin:$PATH"
fi

# set PATH so it includes user's private bin if it exists
if [ -d "$HOME/.local/bin" ]; then
    PATH="$HOME/.local/bin:$PATH"
fi

. "$HOME/.local/bin/env"

# >>> JVM installed by coursier >>>
export JAVA_HOME="/home/sypoo/.cache/coursier/arc/https/github.com/adoptium/temurin11-binaries/releases/download/jdk-11.0.28%252B6/OpenJDK11U-jdk_x64
_linux_hotspot_11.0.28_6.tar.gz/jdk-11.0.28+6"
export PATH="$PATH:/home/sypoo/.cache/coursier/arc/https/github.com/adoptium/temurin11-binaries/releases/download/jdk-11.0.28%252B6/OpenJDK11U-jdk_x6
4_linux_hotspot_11.0.28_6.tar.gz/jdk-11.0.28+6/bin"
# <<< JVM installed by coursier <<<

# >>> coursier install directory >>>
export PATH="$PATH:/home/sypoo/.local/share/coursier/bin"
# <<< coursier install directory <<<
sypoo@pop-os:~/mtuci/mtuci-scala$ 

```

Рис. 3 – изменения в .profile.

```

sypoo@pop-os:~/mtuci/mtuci-scala$ source ~/.profile
sypoo@pop-os:~/mtuci/mtuci-scala$ scala -version
Scala code runner version: 1.9.0
Scala version (default): 3.7.3
sypoo@pop-os:~/mtuci/mtuci-scala$ which scala
/home/sypoo/.local/share/coursier/bin/scala
sypoo@pop-os:~/mtuci/mtuci-scala$ which sbt
/home/sypoo/.local/share/coursier/bin/sbt
sypoo@pop-os:~/mtuci/mtuci-scala$ 

```

Рис. 4 – проверка установки.

Задание 2. Создать простое Scala-приложение через sbt и запустить его из терминала.

```
sypoo@pop-os:~/mtuci/mtuci-scala/lab1$ mkdir hello_world
sypoo@pop-os:~/mtuci/mtuci-scala/lab1$ cd hello_world/
sypoo@pop-os:~/mtuci/mtuci-scala/lab1/hello_world$ ls
sypoo@pop-os:~/mtuci/mtuci-scala/lab1/hello_world$ scala -version
Scala code runner version: 1.9.0
Scala version (default): 3.7.3
sypoo@pop-os:~/mtuci/mtuci-scala/lab1/hello_world$ nvim hello.scala
sypoo@pop-os:~/mtuci/mtuci-scala/lab1/hello_world$ cat hello.scala
//> using scala 3.7.3

@main
def hello(): Unit =
  println("Hello, World!")
```

Рис. 5 – создание простого приложения без sbt.

```
sypoo@pop-os:~/mtuci/mtuci-scala/lab1/hello_world$ scala run hello.scala
Hello, World!
sypoo@pop-os:~/mtuci/mtuci-scala/lab1/hello_world$ □
```

Рис. 6 – вывод простого приложения.

```
sypoo@pop-os:~/mtuci/mtuci-scala/lab1/hello_world_sbt$ mkdir src
sypoo@pop-os:~/mtuci/mtuci-scala/lab1/hello_world_sbt$ cd src/
sypoo@pop-os:~/mtuci/mtuci-scala/lab1/hello_world_sbt/src$ mkdir -p main/scala/example
sypoo@pop-os:~/mtuci/mtuci-scala/lab1/hello_world_sbt/src$ cd main/scala/example
sypoo@pop-os:~/mtuci/mtuci-scala/lab1/hello_world_sbt/src/main/scala/example$ ls
sypoo@pop-os:~/mtuci/mtuci-scala/lab1/hello_world_sbt/src/main/scala/example$ nvim Hello.scala
sypoo@pop-os:~/mtuci/mtuci-scala/lab1/hello_world_sbt/src/main/scala/example$ cat Hello.scala
package example

object Hello {
  def main(args: Array[String]): Unit = {
    println("Hello, World!")
  }
}
sypoo@pop-os:~/mtuci/mtuci-scala/lab1/hello_world_sbt/src/main/scala/example$ nvim Hello.scala
sypoo@pop-os:~/mtuci/mtuci-scala/lab1/hello_world_sbt/src/main/scala/example$ cat Hello.scala
package example

object Hello {
  def main(args: Array[String]): Unit = {
    println("Hello, World!")
  }
}
sypoo@pop-os:~/mtuci/mtuci-scala/lab1/hello_world_sbt/src/main/scala/example$ □
```

Рис. 7 – код приложения через sbt.

```

sypoo@pop-os:~/mtuci/mtuci-scala/lab1/hello_world_sbt$ sbt
[info] welcome to sbt 1.11.7 (Eclipse Adoptium Java 11.0.28)
[info] loading project definition from /home/sypoo/mtuci/mtuci-scala/lab1/hello_world_sbt/project
[info] loading settings for project hello_world_sbt from build.sbt...
[info] set current project to hello_world_sbt (in build file:/home/sypoo/mtuci/mtuci-scala/lab1/hello_world_sbt/)
[info] sbt server started at local:///home/sypoo/.sbt/1.0/server/25ab0cb23b9a080de447/sock
[info] started sbt server
sbt:hello_world_sbt> ~compile
[success] Total time: 0 s, completed Oct 29, 2025, 4:03:03 PM
[info] 1. Monitoring source files for hello_world_sbt/compile...
[info]   Press <enter> to interrupt or '?' for more options.
[info] Build triggered by /home/sypoo/mtuci/mtuci-scala/lab1/hello_world_sbt/src/main/scala/example/Hello.scala. Running 'compile'.
[info] compiling 1 Scala source to /home/sypoo/mtuci/mtuci-scala/lab1/hello_world_sbt/target/scala-2.12/classes ...
https://repo1.maven.org/maven2/org/scala-sbt/util-interface/1.11.5/util-interface-1.11.5.pom
  100.0% [#####] 2.5 KiB (4.6 KiB / s)
https://repo1.maven.org/maven2/org/scala-sbt/compiler-bridge_2.12/1.11.0/compiler-bridge_2.12-1.11.0-sources.jar
  100.0% [#####] 57.1 KiB (483.8 KiB / s)
[info] Non-compiled module 'compiler-bridge_2.12' for Scala 2.12.20. Compiling...
[info]   Compilation completed in 6.384s.
[error] /home/sypoo/mtuci/mtuci-scala/lab1/hello_world_sbt/src/main/scala/example/Hello.scala:4:24: not found: type string
[error]   def main(args: Array[String]): Unit = {
[error]
[error] one error found
[error] (Compile / compileIncremental) Compilation failed
[error] Total time: 8 s, completed Oct 29, 2025, 4:06:33 PM
[info] 2. Monitoring source files for hello_world_sbt/compile...
[info]   Press <enter> to interrupt or '?' for more options.
[info] Build triggered by /home/sypoo/mtuci/mtuci-scala/lab1/hello_world_sbt/src/main/scala/example/Hello.scala. Running 'compile'.
[info] compiling 1 Scala source to /home/sypoo/mtuci/mtuci-scala/lab1/hello_world_sbt/target/scala-2.12/classes ...
[success] Total time: 1 s, completed Oct 29, 2025, 4:07:15 PM
[info] 3. Monitoring source files for hello_world_sbt/compile...
[info]   Press <enter> to interrupt or '?' for more options.
[info] Received input event: CancelWatch.
sbt:hello_world_sbt> run
[info] running example.Hello
Hello, World!
[success] Total time: 0 s, completed Oct 29, 2025, 4:07:38 PM
sbt:hello_world_sbt> 

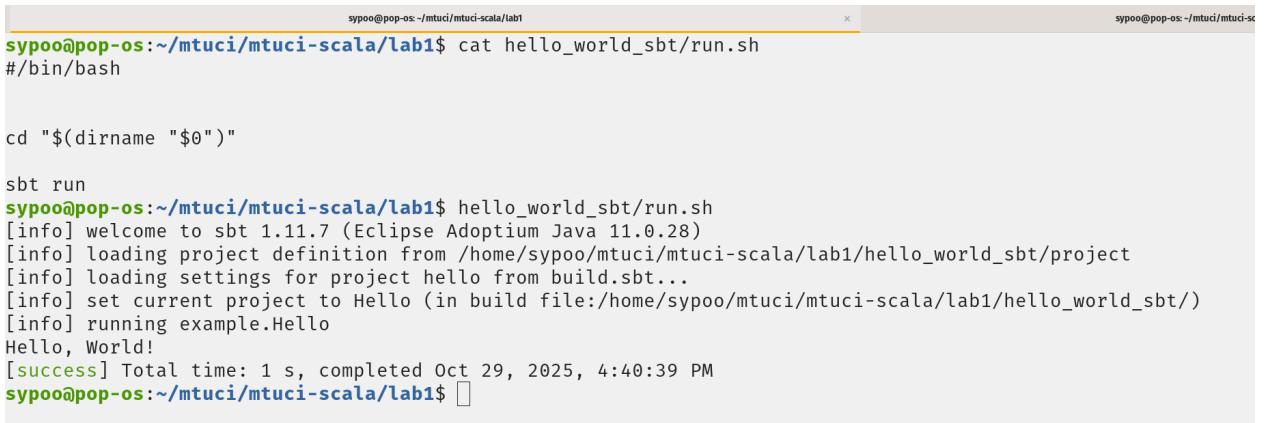
```

Рис. 8 – результат.

Задание 3. Настроить переменные окружения (PATH и др.).

Рис. 9 – проверка местонахождения исполняемых файлов.

Задание 4. Создать bash-скрипт для автоматического запуска Scala-программы.



```
sypoo@pop-os:~/mtuci/mtuci-scala/lab1$ cat hello_world_sbt/run.sh
#!/bin/bash

cd "$(dirname "$0")"
sbt run
sypoo@pop-os:~/mtuci/mtuci-scala/lab1$ hello_world_sbt/run.sh
[info] welcome to sbt 1.11.7 (Eclipse Adoptium Java 11.0.28)
[info] loading project definition from /home/sypoo/mtuci/mtuci-scala/lab1/hello_world_sbt/project
[info] loading settings for project hello from build.sbt...
[info] set current project to Hello (in build file:/home/sypoo/mtuci/mtuci-scala/lab1/hello_world_sbt/)
[info] running example.Hello
Hello, World!
[success] Total time: 1 s, completed Oct 29, 2025, 4:40:39 PM
sypoo@pop-os:~/mtuci/mtuci-scala/lab1$
```

Рис. 10 – создание run.sh для запуска приложения одной командой из любого места.

Индивидуальное задание. Написать скрипт установки Scala и sbt.

```
install.scala_and_sbt.sh ×
1  #!/bin/bash
2
3  # Функция для автоматического ответа "Y" на все вопросы
4  auto_yes() {
5      yes "" | "$@" || return 1
6  }
7
8  # Функция для вывода сообщений об ошибках
9  error_exit() {
10     echo "$1" 1>&2
11     exit 1
12 }
13
14 # Проверка архитектуры процессора
15 ARCH=$(uname -m)
16 case "$ARCH" in
17     x86_64)
18         COURSIER_URL="https://github.com/coursier/coursier/releases/latest/download/cs-x86_64-pc-linux.gz"
19         ;;
20     aarch64|arm64)
21         COURSIER_URL="https://github.com/VirtusLab/coursier-m1/releases/latest/download/cs-aarch64-pc-linux.gz"
22         ;;
23     *)
24         error_exit "Неподдерживаемая архитектура: $ARCH"
25     ;;
26 esac
27
28 # Проверка наличия curl и gzip
29 if ! command -v curl >& /dev/null; then
30     error_exit "curl не установлен. Установите его и повторите попытку."
31 fi
32 if ! command -v gzip >& /dev/null; then
33     error_exit "gzip не установлен. Установите его и повторите попытку."
34 fi
35
36 # Установка Coursier
37 echo "Установка Coursier..."
38 curl -fL "$COURSIER_URL" | gzip -d > cs || error_exit "Не удалось скачать или распаковать Coursier."
39 chmod +x cs || error_exit "Не удалось сделать cs исполняемым."
```

Рис. 11 – код скрипта часть 1.

```
install.scala_and_sbt.sh ×
35 # Установка Coursier
36 echo "Установка Coursier..."
37 curl -fL "$COURSIER_URL" | gzip -d > cs || error_exit "Не удалось скачать или распаковать Coursier."
38 chmod +x cs || error_exit "Не удалось сделать cs исполняемым."
39
40 # Автоматическая установка с подтверждением всех шагов
41 echo "Запуск coursier setup с автоматическими ответами..."
42 auto_yes ./cs setup || error_exit "Не удалось выполнить coursier setup."
43
44 # Очистка временного файла
45 rm -f cs
46
47 # Активация изменений в ~/.profile
48 echo "Активация изменений в ~/.profile..."
49 source ~/.profile || error_exit "Не удалось активировать ~/.profile."
50
51 # Проверка установки
52 echo "Проверка установки..."
53 if ! command -v scala >& /dev/null; then
54     error_exit "Scala не установлена."
55 fi
56 if ! command -v sbt >& /dev/null; then
57     error_exit "sbt не установлен."
58 fi
59
60 echo "Scala и sbt успешно установлены через Coursier!"
61 echo "Все изменения активированы. Вы можете использовать Scala и sbt прямо сейчас."
```

Рис. 12 – код скрипта часть 2.

```
hotspot_11.0.28_6.tar.gz (97.82 %, 191138626 / 195406004)

Downloaded https://github.com/adoptium/temurin11-binaries/releases/download/jdk-11.0.28%2B6/OpenJDK11U-jdk
_x64_linux_hotspot_11.0.28_6.tar.gz
Should we update ~/.profile? [Y/n] Some shell configuration files were updated. It is recommended to close
this terminal once the setup command is done, and open a new one for the changes to be taken into account.

Checking if ~/.local/share/coursier/bin is in PATH
Should we add ~/.local/share/coursier/bin to your PATH via ~/.profile? [Y/n]
Checking if the standard Scala applications are installed
  Installed ammonite
  Installed cs
  Installed coursier
  Installed scala
  Installed scalac
  Installed scala-cli
  Installed sbt
  Installed sbtn
  Installed scalafmt

Активация изменений в ~/.profile...
Проверка установки...
Scala и sbt успешно установлены через Coursier!
Все изменения активированы. Вы можете использовать Scala и sbt прямо сейчас.
```

Рис. 13 – результат работы скрипта.

Заключение

Вывод: проделав работу, мы познакомились с основными командами операционной системы Linux, освоили работу с терминалом и изучили структуру файловой системы. Научились устанавливать и настраивать окружение для разработки на языке Scala, а также приобрели базовые навыки взаимодействия с консольным интерфейсом и пакетным менеджером.