**Министерство науки и высшего образования Российской Федерации**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**Факультет цифровых трансформаций**

**Дисциплина:**

«Архитектура вычислительных сетей»

**ПРАКТИЧЕСКАЯ РАБОТА №3**

«Разработка скриптов на BASH»

**Выполнил:**

Мавров Артём Николаевич

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

(подпись)

**Проверила:**

Александрова С.А., преподаватель

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

(отметка о выполнении)

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

(подпись)

Содержание

[Содержание 2](#_Toc177380279)

[Постановка задачи 3](#_Toc177380280)

[Артефакты выполнения 4](#_Toc177380281)

[Ответы на вопросы 5](#_Toc177380282)

[Использование GIA 6](#_Toc177380283)

[Рефлексия 7](#_Toc177380284)

[Список использованных источников 8](#_Toc177380285)

Постановка задачи

**Цель работы** – получить практические навыки программирования на bash, освоить

практические приемы работы с процессами в Linux, закрепить материалы лекций по архитектуре CPU.

Применяемое в ходе выполнения практической работы ПО (если используются команды и утилиты ОС, дайте краткое описание назначения утилит):

1.Virtual Box

2. Debian 12

Артефакты выполнения

**1.1 Команда 1**

****

**1.2 КОМАНДА 2**

****

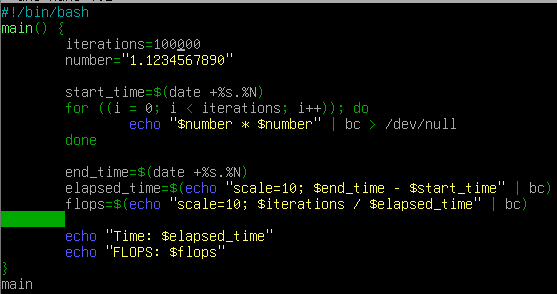
**1.3 Команда 3**

****

**1.4 Команда 4**

****

**1.5 Скрипт 1**

****

**1.6 Скрипт 2**

**Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание**

**1,7 Скрипт 3**

**Изображение выглядит как текст, Шрифт, снимок экрана, дизайн

Автоматически созданное описание**

Ответы на вопросы

**2.1 ВОПРОС 1**

Чтобы увидеть дерево процессов, в htop нажмите клавишу **F5**. Это отобразит процессы в виде иерархии, где родительские процессы находятся выше, а дочерние — ниже.

Процесс становится потомком другого, если запускается из него.

**2.2 ВОПРОС 2**

Топология CPU описывает физическую и логическую организацию процессоров, включая ядра, потоки, сокеты и кэш.

**2.3 ВОПРОС 3**

1)Да

2)Да

**2.4 ВОПРОС 4**

* **Точность измерений:** Мы не учитываем накладные расходы ОС (например, переключение контекста, кеширование).
* **Однопоточность:** Скрипт измеряет только производительность одного потока, а не всего процессора.
* **Тип операций:** Умножение — это простая операция. Другие математические задачи (например, деление) могут использовать больше ресурсов.
* **Неравномерность загрузки:** Фоновая активность системы может влиять на результаты.

**2.5 ВОПРОС 5**

* Процесс с высоким приоритетом (nice -n -5) получал больше CPU-времени, чем процесс с низким приоритетом (nice -n 10).
* Даже при высокой нагрузке утилитой stress процесс с высоким приоритетом продолжал получать большую долю процессорного времени.
* При низком приоритете процесс с математическими вычислениями практически не влиял на производительность системы.

**2.6 ВОПРОС 6**

* **Изменение привязки ядра (taskset):** Привязал процессы к разным ядрам для уменьшения конкуренции.
* **Изменение приоритетов (renice):** Установил более высокий приоритет для ресурсоёмкого процесса.
* **Результат:**
* Процесс с математическими вычислениями стал менее конкурировать с процессами чтения/записи.
* Нагрузка на систему распределилась равномернее, увеличив производительность.

Использование GIA

GIA был использован для ускорения процесса поиска справочных материалов.

РЕфлексия

Выполнение заданий дало возможность глубже понять принципы управления процессами и вычислительными ресурсами в Linux. Основные выводы:

1. **Управление процессами:** Работа с htop, ps, kill и taskset показала, как можно контролировать выполнение процессов, их приоритеты и привязку к ядрам. Это оказалось полезным для оптимизации нагрузки на систему.
2. **Точность измерений:** Создание скрипта для расчёта FLOPS помогло понять ограничения простых методов оценки производительности процессора. Точность результатов зависит от учёта множества факторов, таких как системные накладные расходы и использование многопоточности.
3. **Нагрузочное тестирование:** Использование stress и работа с приоритетами позволили проанализировать, как операционная система распределяет ресурсы между процессами. Стало очевидно, что правильная настройка приоритетов и привязка процессов к ядрам значительно влияют на производительность.
4. **Практическая ценность:** Все изученные команды и концепции полезны для администрирования, диагностики и оптимизации производительности систем. Эти навыки применимы как для локальных серверов, так и для виртуальных машин в облаке.

Проделанная работа была полезной и показала, как взаимодействие с ОС на низком уровне помогает решать задачи производительности и устойчивости системы

Список использованных источников

1. Intel kernel. Assembler. // Режим доступа: [ссылка1](http://google.com/)

(даты обращения: 11.11.2024-19.11.2024)

2. Linux Auditd Best Practice Configuration // Github [Электронный ресурс]. Режим доступа: [ссылка](http://goog.com/)2

(дата обращения: 19.11.2024)