|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

*ФАКУЛЬТЕТ «Информатика и системы управления»*

*КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»*

**Отчет**

|  |  |
| --- | --- |
| **по лабораторной работе №** | 03 |

**Название:**

***Командная строка ОС UNIX/LINUX***

**Дисциплина:  *Операционные системы***

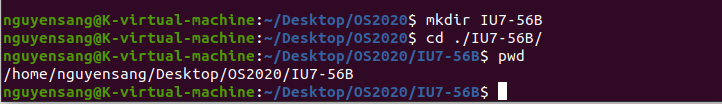
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ***ИУ7И-56Б*** |  |  | **Нгуен Ф. С.** |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | **Рязанова Н. Ю.** |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

*Москва, 2020*

* **Задание 1**
* Используя команду mkdir cоздайте директорию имeнем своей группы.

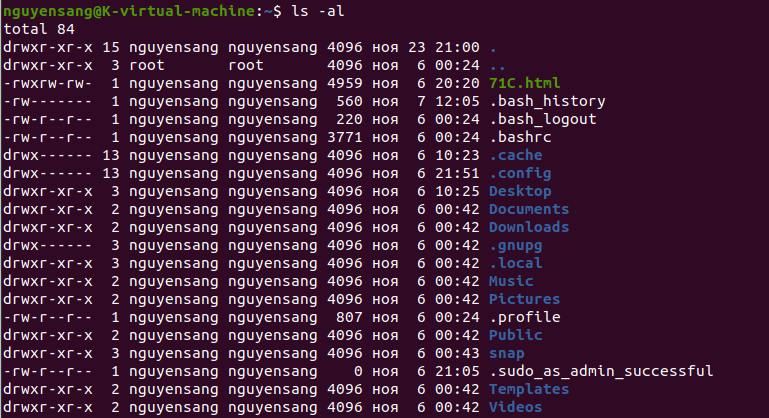
$ **mkdir IU7-56B**

**$ cd ./IU7-56B**



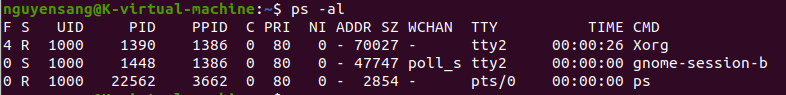
* **Команда: ls**

**$ ls –al**

****

* **Команда: ps**

**$ ps –al**

****

* **Задание 2**

Напишите программу, в которой создается дочерний процесс и организуйте как в предке, так и в потомке бесконечные циклы, в которых выводятся идентификаторы процессов с помощью системного вызова getpid():

1. #include <stdio.h>
2. **int** main()
3. {
4. **int** childpid;
5. **if** ((childpid = fork())== -1)
6. {
7. perror(“Can’t fork.\n”);
8. **return** 1;
9. }
10. **else** **if** (childpid == 0)
11. {
12. **while** (1) printf(“ %d  “, getpid());
13. **return** 0;
14. }
15. **else**
16. {
17. **while**(1)  printf(“ %d  “,getpid());
18. **return** 0;
19. }
20. }

* запустите программу и посмотрите идентификаторы созданных процессов: предка и потомка;

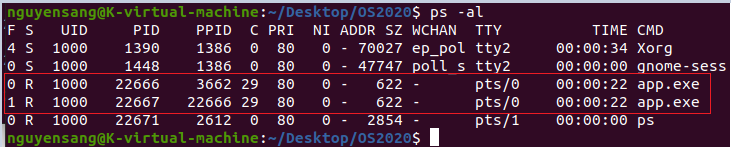
**$ gcc –std=c99 –o main.c app.exe**

**$ ./app.exe**

**$ ps –al**

Предка: UID = 22666

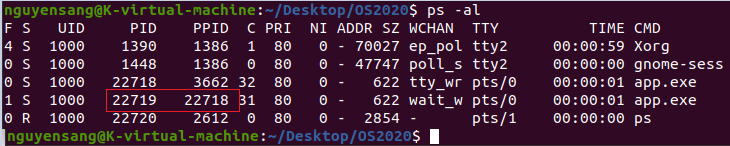
Потомка: UID = 22667



* Процесс зомби

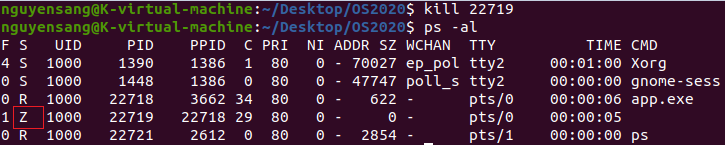
**$ ./app.exe**

**$ ps –al**



**$ kill 22719**

**$ ps –al**



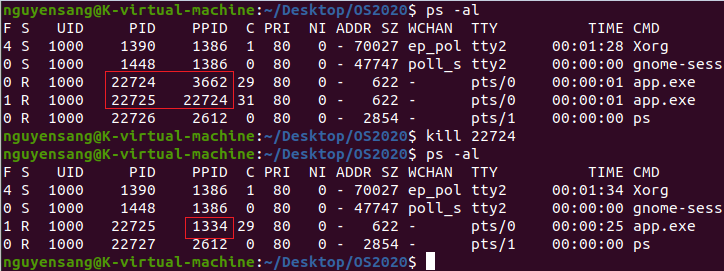
**$ kill 22718**

* Получениe «осиротевшего» процесса

**$ ./app.exe**

**$ ls –al**

**$ kill 22724**

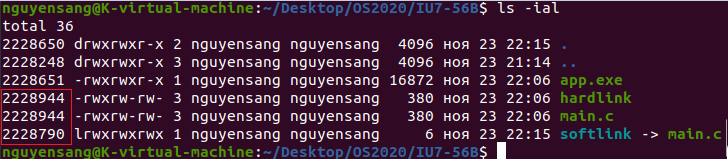


* **Задание 3**: Жесткие и гибкие ссылки на файлы

**$ ln main.c hardlink**

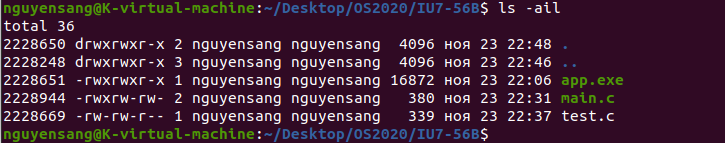
**$ ln –s main softlink**

**$ ls –ial**

****

**$ rm hardlink**

**$ rm softlink**



**Задание:** написать программу вывода на экран сообщения «Hellow, world!».

1. #include <stdio.h>
2. **int** main (**void**)
3. {
4. printf(“Hello, World!\n”);
5. **return** 0;
6. }

**vi helloworld.c**

Нажмите клавишу I, чтобы вставить

Клавиша Esc для выхода из вставки

**:w** – сохранять

**:q** - Выход



**gcc –std=c99 –o helloworld.exe helloworld.c**

**./helloworld.exe > ./a.out**

**tee < ./a.out**

