



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«МИРЭА – Российский технологический университет»
РТУ МИРЭА

Институт кибербезопасности и цифровых технологий
Кафедра КБ-14 «Цифровые технологии обработки данных»

ОТЧЁТ ПО ПРАКТИЧЕСКОМУ ЗАНЯТИЮ
по дисциплине «Безопасность операционных систем»

Отчет представлен к
рассмотрению:

Студент группы БСБО-16-23

«18» октября 2024 г.

Крашенинников
М. В.

(подпись)

Преподаватель

«18» октября 2024 г.

Иванова И. А.

(подпись)

Москва, 2024 г.

Задание 1.

```
#include <iostream>
#include <unistd.h>

int main() {
    std::cout << "PID: " << getpid() << std::endl;
    std::cout << "PPID: " << getppid() << std::endl;

    return 0;
}
```

Вывод программы:

```
twox@Matvey:/mnt/c/Users/boltf/OneDrive/Рабочий стол/Безопасность ОС/практика 3$ g++ -Wall -Wextra -Werror task1.cpp
twox@Matvey:/mnt/c/Users/boltf/OneDrive/Рабочий стол/Безопасность ОС/практика 3$ ./a.out
PID: 15210
PPID: 1865
twox@Matvey:/mnt/c/Users/boltf/OneDrive/Рабочий стол/Безопасность ОС/практика 3$ ps grep 15210
  PID TTY          STAT       TIME COMMAND
 15231 pts/4        R+          0:00 ps grep 15210 SHELL=/bin/bash WSL2_GUI_APPS_ENABLED=1 WSL_DISTRO_NAME=Ubuntu NAME=Matvey PW
twox@Matvey:/mnt/c/Users/boltf/OneDrive/Рабочий стол/Безопасность ОС/практика 3$ ps grep 1865
  PID TTY          STAT       TIME COMMAND
 15249 pts/4        R+          0:00 ps grep 1865 SHELL=/bin/bash WSL2_GUI_APPS_ENABLED=1 WSL_DISTRO_NAME=Ubuntu NAME=Matvey PWD
twox@Matvey:/mnt/c/Users/boltf/OneDrive/Рабочий стол/Безопасность ОС/практика 3$ |
```

Задание 2.

```
1  import multiprocessing
2  import time
3
4  sema_point1 = multiprocessing.Semaphore(0) # Для В и С
5  sema_point2 = multiprocessing.Semaphore(0) # Для D, E, F
6  sema_point3 = multiprocessing.Semaphore(0) # Для Г и Н
7  sema_point4 = multiprocessing.Semaphore(0) # Для К
8
9  def process_a():
10     print("Процесс 'A' начался и проходит точку 1.")
11     time.sleep(1)
12     sema_point1.release()
13     sema_point1.release()
14     print("Процесс 'A' завершил точку 1 и переходит к точке 2.")
15     time.sleep(1)
16     sema_point2.release()
17     sema_point2.release()
18
19  def process_b():
20     sema_point1.acquire()
21     print("Процесс 'B' проходит точку 1.")
22     time.sleep(1)
```

```
24 def process_c():
25     sema_point1.acquire()
26     print("Процесс 'C' проходит точку 1.")
27     time.sleep(1)
28     sema_point2.release()
29
30 def process_d():
31     sema_point2.acquire()
32     print("Процесс 'D' проходит точку 2.")
33     time.sleep(1)
34
35 def process_e():
36     sema_point2.acquire()
37     print("Процесс 'E' проходит точку 2.")
38     time.sleep(1)
39
40 def process_f():
41     sema_point2.acquire()
42     print("Процесс 'F' проходит точку 2.")
43     time.sleep(1)
44     sema_point3.release()
45     sema_point3.release()
```

```
47 def process_g():
48     sema_point3.acquire()
49     print("Процесс 'G' проходит точку 3.")
50     time.sleep(1)
51
52 def process_h():
53     sema_point3.acquire()
54     print("Процесс 'H' проходит точку 3.")
55     time.sleep(1)
56     sema_point4.release()
57
58 def process_k():
59     sema_point4.acquire()
60     print("Процесс 'K' завершает все в точке 4.")
61     time.sleep(1)
62
63 def main():
64     processes = [
65         multiprocessing.Process(target=process_a),
66         multiprocessing.Process(target=process_b),
67         multiprocessing.Process(target=process_c),
68         multiprocessing.Process(target=process_d),
69         multiprocessing.Process(target=process_e),
70         multiprocessing.Process(target=process_f),
71         multiprocessing.Process(target=process_g),
72         multiprocessing.Process(target=process_h),
73         multiprocessing.Process(target=process_k),
74     ]
75
```

```

76         for p in processes:
77             p.start()
78
79         for p in processes:
80             p.join()
81
82     if __name__ == "__main__":
83         main()

```

Вывод программы:

```

twox@Matvey:/mnt/c/Users/boltf/OneDrive/Рабочий стол/Безопасность ОС/практика 3$ python3 task2.py
Процесс 'A' начался и проходит точку 1.
Процесс 'A' завершил точку 1 и переходит к точке 2.
Процесс 'C' проходит точку 1.
Процесс 'B' проходит точку 1.
Процесс 'D' проходит точку 2.
Процесс 'F' проходит точку 2.
Процесс 'E' проходит точку 2.
Процесс 'G' проходит точку 3.
Процесс 'H' проходит точку 3.
Процесс 'K' завершает все в точке 4.

```

Задание 3.

```

1  import os
2  import time
3
4  def child_process():
5      print(f"Дочерний процесс с PID: {os.getpid()}, родительский PID: {os.getppid()}")
6      time.sleep(5)
7      print(f"Дочерний процесс с PID: {os.getpid()} теперь сирота, его родительский PID: {os.getppid()}")
8
9  def main():
10     pid = os.fork()
11
12     if pid > 0:
13         print(f"Родительский процесс с PID: {os.getpid()}")
14         print(f"Создан дочерний процесс с PID: {pid}")
15         time.sleep(2)
16         print("Родительский процесс завершает работу")
17     else:
18         child_process()
19
20 if __name__ == "__main__":
21     main()

```

Вывод программы:

```

twox@Matvey:/mnt/c/Users/boltf/OneDrive/Рабочий стол/Безопасность ОС/практика 3$ python3 task3.py
Дочерний процесс с PID: 10600, родительский PID: 10599
Родительский процесс с PID: 10599
Создан дочерний процесс с PID: 10600
Родительский процесс завершает работу
twox@Matvey:/mnt/c/Users/boltf/OneDrive/Рабочий стол/Безопасность ОС/практика 3$ Дочерний процесс с PID: 10600 теперь си
рота, его родительский PID: 1863
^C

```


Задание 4.

```
1  import threading
2
3  def thread_function(param1, param2, param3):
4      print(f"Параметр 1: {param1}")
5      print(f"Параметр 2: {param2}")
6      print(f"Параметр 3: {param3}")
7
8  if __name__ == "__main__":
9      param1 = 666
10     param2 = 3.14
11     param3 = "HELLO"
12
13     thread = threading.Thread(target=thread_function, args=(param1, param2, param3))
14     thread.start()
15     thread.join()
16
17     print("Поток завершен")
```

Вывод программы:

```
twox@Matvey:/mnt/c/Users/boltf/OneDrive/Рабочий стол/Безопасность ОС/практика 3$ python3 task4.py
Параметр 1: 666
Параметр 2: 3.14
Параметр 3: HELLO
Поток завершен
```

Задание 5.

```
1  import threading
2  import time
3
4  readers = 0
5  mutex = threading.Semaphore(1)
6  room_empty = threading.Semaphore(1)
7  shared_data = 0
8
9  def reader(reader_id):
10     global readers
11     while True:
12         mutex.acquire()
13         readers += 1
14         if readers == 1:
15             room_empty.acquire()
16         mutex.release()
17
18         print(f"Читатель {reader_id} читает данные: {shared_data}")
19         time.sleep(1)
20
21         mutex.acquire()
22         readers -= 1
23         if readers == 0:
24             room_empty.release()
25         mutex.release()
26
27         time.sleep(2)
```

```

29 def writer(writer_id):
30     global shared_data
31     while True:
32         room_empty.acquire()
33         shared_data += 1
34         print(f"Писатель {writer_id} изменил данные на: {shared_data}")
35         time.sleep(2)
36         room_empty.release()
37
38         time.sleep(3)
39
40 def main():
41     for i in range(3):
42         threading.Thread(target=reader, args=(i,)).start()
43
44         threading.Thread(target=writer, args=(1,)).start()
45
46 if __name__ == "__main__":
47     main()
48

```

Вывод программы:

```

twox@Matvey:/mnt/c/Users/boltf/OneDrive/Рабочий стол/Безопасность ОС/практика 3$ python3 task5.py
Читатель 0 читает данные: 0
Читатель 1 читает данные: 0
Читатель 2 читает данные: 0
Писатель 1 изменил данные на: 1
Читатель 0 читает данные: 1
Читатель 1 читает данные: 1
Читатель 2 читает данные: 1
Писатель 1 изменил данные на: 2
Читатель 0 читает данные: 2
Читатель 1 читает данные: 2
Читатель 2 читает данные: 2
Читатель 0 читает данные: 2
Читатель 1 читает данные: 2
Читатель 2 читает данные: 2
Писатель 1 изменил данные на: 3
Читатель 0 читает данные: 3
Читатель 1 читает данные: 3
Читатель 2 читает данные: 3
Писатель 1 изменил данные на: 4
Читатель 2 читает данные: 4
Читатель 0 читает данные: 4
Читатель 1 читает данные: 4
Писатель 1 изменил данные на: 5
Читатель 2 читает данные: 5
Читатель 1 читает данные: 5
Читатель 0 читает данные: 5
Писатель 1 изменил данные на: 6
Читатель 2 читает данные: 6
Читатель 1 читает данные: 6

```