Үйлдлийн системийн онол
Семинар 8: Page replacement algorithm
Г. Саруул 20B1NUM2095

```python
import sys

def FIFO(size, pages):
    SIZE = size
    count = 0
    memory = []
    faults = 0
    fifoIndex = 0

    for page in pages:
        if memory.count(page) == 0 and count < SIZE:
            memory.append(page)
            count += 1
            faults += 1
        elif memory.count(page) == 0 and count == SIZE:
            memory[fifoIndex] = page
            fifoIndex = (fifoIndex + 1) % SIZE
            faults += 1
        elif memory.count(page) > 0:
            pass
    return faults

def LRU(size, pages):
    SIZE = size
    count = 0
    memory = []
    faults = 0
    for page in pages:
        if memory.count(page) == 0 and count < SIZE:
            memory.append(page)
            count += 1
            faults += 1
        elif memory.count(page) == 0 and count == SIZE:
            memory.pop(0)
            memory.append(page)
            faults += 1
        elif memory.count(page) > 0:
            memory.remove(page)
            memory.append(page)
```

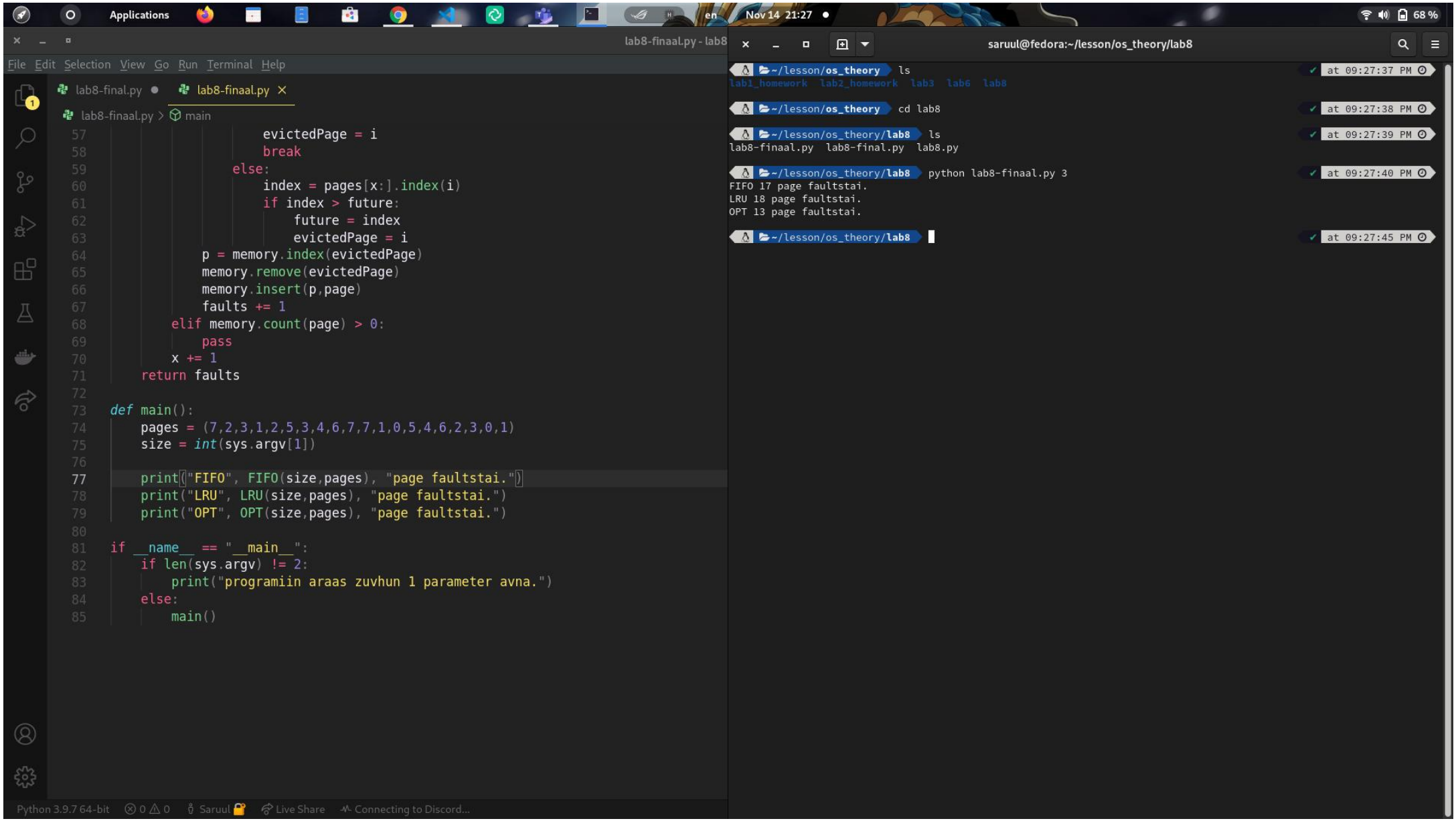
```
    return faults

def OPT(size,pages):
    SIZE = size
    count = 0
    memory = []
    faults = 0
    x = 0
    for page in pages:
        if memory.count(page) == 0 and count < SIZE:
            memory.append(page)
            count += 1
            faults += 1
        elif memory.count(page) == 0 and count == SIZE:
            future = -1
            for i in memory:
                if pages[x:].count(i) == 0:
                    evictedPage = i
                    break
                else:
                    index = pages[x:].index(i)
                    if index > future:
                        future = index
                        evictedPage = i
            p = memory.index(evictedPage)
            memory.remove(evictedPage)
            memory.insert(p,page)
            faults += 1
        elif memory.count(page) > 0:
            pass
        x += 1
    return faults

def main():
    pages = (7,2,3,1,2,5,3,4,6,7,7,1,0,5,4,6,2,3,0,1)
    size = int(sys.argv[1])

    print("FIFO", FIFO(size,pages), "page faulttai.")
    print("LRU", LRU(size,pages), "page faulttai.")
```

```
73  def main():
74      pages = (7,2,3,1,2,5,3,4,6,7,7,1,0,5,4,6,2,3,0,1)
75      size = int(sys.argv[1])
76
77      print("FIFO", FIFO(size,pages), "page faulttai.")
78      print("LRU", LRU(size,pages), "page faulttai.")
79      print("OPT", OPT(size,pages), "page faulttai.")
80
81  if __name__ == "__main__":
82      if len(sys.argv) != 2:
83          print("programiin araas zuvhun 1 parameter avna.")
84      else:
85          main()
```


Applications
en
Nov 14 21:27
68 %
lab8-finaal.py - lab8
File Edit Selection View Go Run Terminal Help
lab8-final.py
lab8-finaal.py
lab8-finaal.py > main
evictedPage = i
break
else:
index = pages[x:].index(i)
if index > future:
future = index
evictedPage = i
p = memory.index(evictedPage)
memory.remove(evictedPage)
memory.insert(p,page)
faults += 1
elif memory.count(page) > 0:
pass
x += 1
return faults
def main():
pages = (7,2,3,1,2,5,3,4,6,7,7,1,0,5,4,6,2,3,0,1)
size = int(sys.argv[1])
print("FIFO", FIFO(size,pages), "page faultstai.")
print("LRU", LRU(size,pages), "page faultstai.")
print("OPT", OPT(size,pages), "page faultstai.")
if __name__ == "__main__":
if len(sys.argv) != 2:
print("programiin araas zuvhun 1 parameter avna.")
else:
main()
Python 3.9.7 64-bit
Saruul
Live Share
Connecting to Discord...
saruul@fedora:~/lesson/os_theory/lab8
~/lesson/os_theory ls
at 09:27:37 PM
lab1_homework lab2_homework lab3 lab6 lab8
~/lesson/os_theory cd lab8
at 09:27:38 PM
~/lesson/os_theory/lab8 ls
at 09:27:39 PM
lab8-finaal.py lab8-final.py lab8.py
~/lesson/os_theory/lab8 python lab8-finaal.py 3
at 09:27:40 PM
FIFO 17 page faultstai.
LRU 18 page faultstai.
OPT 13 page faultstai.
~/lesson/os_theory/lab8
at 09:27:45 PM