

SISTEM OPERASI
Unix System Call dan Manajemen Memori
Lavina 2341760062

Tugas Pendahuluan

1. Apa yang dimaksud dengan system call ?

Jawab : System call adalah cara bagi program aplikasi untuk meminta layanan dari kernel sistem operasi. Kernel adalah bagian inti dari sistem operasi yang bertanggung jawab atas manajemen sumber daya sistem, seperti memori, proses, dan perangkat keras.

2. Apa yang dimaksud dengan sistem call fork(), execl() dan wait(). Jawablah dengan menggunakan perintah man (contoh : man 2 fork, man 2 execl dan man 2 wait) ?

Jawab : System call **fork()** adalah suatu system call yang membuat suatu proses baru pada system operasi UNIX cara menggunakannya yaitu: **man 2 fork**. Sistem call **execl()** meletakkan program executable baru ke memory dan mengasosiasikannya dengan proses saat itu caramenggunakannya yaitu: **man 2 execl**. System call **wait()** menyebabkan proses menunggu sinyal (menunggu sampai sembarang tipe sinyal diterima dari sembarang proses) cara menggunakannya yaitu: **man 2 wait**.

3. Apa yang dimaksud sistem virtual memory, proses swapping dan buffer cache pada manajemen memory ?

Jawab : **Sistem virtual memory** adalah teknik manajemen memori yang memungkinkan sistem operasi untuk membuat memori virtual yang lebih besar daripada memori fisik yang tersedia. **Swapping** adalah proses memindahkan halaman memori dari disk ke memori fisik, atau dari memori fisik ke disk. **Buffer cache** adalah area memori yang digunakan untuk menyimpan data yang sering diakses.

4. Apa yang dimaksud perintah free dan cat /proc/meminfo ?

Jawab : Perintah **free** digunakan untuk menampilkan informasi tentang penggunaan memori sistem. Informasi ini termasuk jumlah memori fisik yang tersedia, jumlah memori fisik yang digunakan, dan jumlah memori virtual yang digunakan. **Perintah cat**

/proc/meminfo digunakan untuk menampilkan informasi terperinci tentang penggunaan memori sistem.

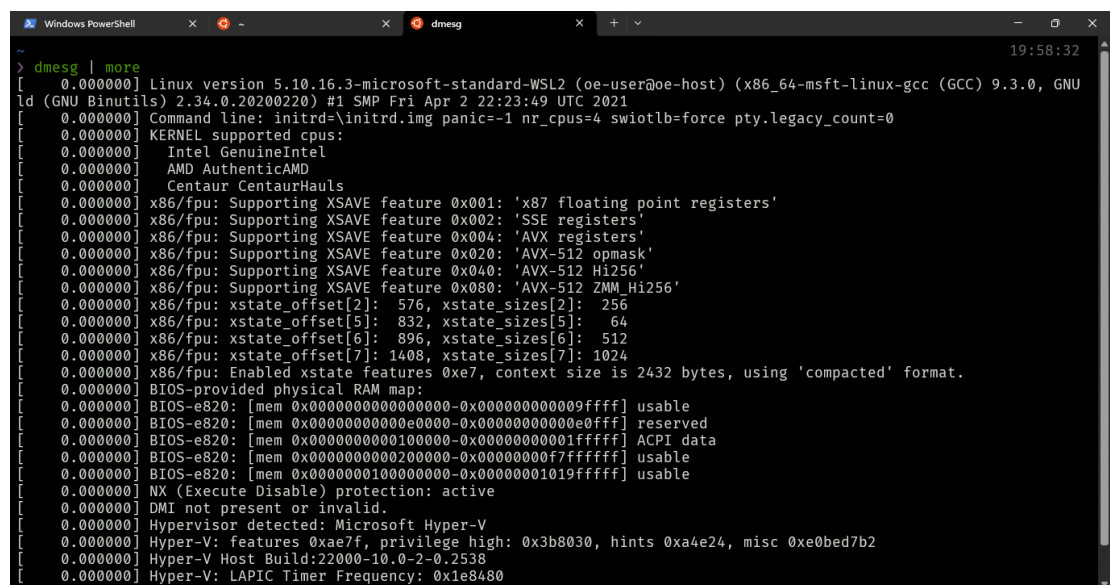
5. Apa yang dimaksud perintah **ps** ?

Jawab : Perintah **ps** digunakan untuk menampilkan informasi tentang proses yang sedang berjalan. Informasi ini termasuk ID proses (PID), nama proses, status proses, dan penggunaan memori proses.

Praktikum 7

1. Perhatikan dengan perintah **dmesg** jumlah memory tersedia dan proses swapping

\$ dmesg | more

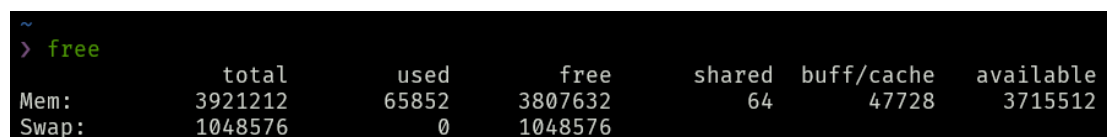


```
> dmesg | more
[ 0.000000] Linux version 5.10.16.3-microsoft-standard-WSL2 (oe-user@oe-host) (x86_64-msft-linux-gcc (GCC) 9.3.0, GNU
ld (GNU Binutils) 2.34.0.20200220) #1 SMP Fri Apr 2 22:23:49 UTC 2021
[ 0.000000] Command line: initrd=\initrd.img panic=-1 nr_cpus=4 swiotlb=force pty.legacy_count=0
[ 0.000000] KERNEL supported cpus:
[ 0.000000]   Intel GenuineIntel
[ 0.000000]   AMD AuthenticAMD
[ 0.000000]   Centaur CentaurHauls
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x001: 'x87 floating point registers'
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x002: 'SSE registers'
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x004: 'AVX registers'
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x020: 'AVX-512 opmask'
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x040: 'AVX-512 Hi256'
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x080: 'AVX-512 ZMM_Hi256'
[ 0.000000] x86/fpu: xstate_offset[2]: 576, xstate_sizes[2]: 256
[ 0.000000] x86/fpu: xstate_offset[5]: 832, xstate_sizes[5]: 64
[ 0.000000] x86/fpu: xstate_offset[6]: 896, xstate_sizes[6]: 512
[ 0.000000] x86/fpu: xstate_offset[7]: 1408, xstate_sizes[7]: 1024
[ 0.000000] x86/fpu: Enabled xstate features 0xe7, context size is 2432 bytes, using 'compacted' format.
[ 0.000000] BIOS-provided physical RAM map:
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000009ffff] usable
[ 0.000000] BIOS-e820: [mem 0x00000000000e0000-0x00000000000e0fff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000000100000-0x00000000001fffff] ACPI data
[ 0.000000] BIOS-e820: [mem 0x0000000000200000-0x00000000007fffff] usable
[ 0.000000] BIOS-e820: [mem 0x0000000010000000-0x000000001019ffff] usable
[ 0.000000] NX (Execute Disable) protection: active
[ 0.000000] DMI not present or invalid.
[ 0.000000] Hypervisor detected: Microsoft Hyper-V
[ 0.000000] Hyper-V: features 0xae7f, privilege high: 0x3b8030, hints 0xa4e24, misc 0xe0bed7b2
[ 0.000000] Hyper-V Host Build:22000-10.0-2-0.2538
[ 0.000000] Hyper-V: LAPIC Timer Frequency: 0x1e8480
```

Perintah **dmesg** berguna untuk menampilkan pesan dari kernel ring buffer, pesannya berisi informasi tentang proses booting, hardware yang terdeteksi dan aktivitas kernel lainnya, informasi tersebut kemudian ditampilkan per halaman dengan perintah **more**.

2. Dengan perintah **free** perhatikan jumlah memory "free", "used", "share" dan "buffer".

\$ free



```
> free
              total        used        free      shared  buff/cache   available
Mem:           3921212       65852       3807632          64        47728       3715512
Swap:          1048576           0          1048576
```

Perintah tersebut menampilkan jumlah memori, memori yang terpakai maupun yang tidak terpakai, jumlah memori virtual (swap), jumlah memori untuk cache buffer.

3. Dengan perintah dibawah ini apakah hasilnya sama dengan no 2 ?

```
$ cat /proc/meminfo
```

```
> cat /proc/meminfo
MemTotal:      3921212 kB
MemFree:       3808932 kB
MemAvailable:  3716044 kB
Buffers:       9712 kB
Cached:        19272 kB
SwapCached:    0 kB
Active:        16224 kB
Inactive:      19204 kB
Active(anon):  84 kB
Inactive(anon): 6416 kB
Active(file):  16140 kB
Inactive(file): 12788 kB
Unevictable:   0 kB
Mlocked:       0 kB
SwapTotal:    1048576 kB
SwapFree:     1048576 kB
Dirty:         336 kB
Writeback:     0 kB
AnonPages:    6368 kB
Mapped:       7760 kB
Shmem:        64 kB
KReclaimable: 17156 kB
Slab:         37544 kB
SReclaimable: 17156 kB
SUnreclaim:   20388 kB
KernelStack:  1792 kB
PageTables:    400 kB
NFS_Unstable:  0 kB
Bounce:       0 kB
```

Pada perintah `free` sistem menampilkan informasi dalam format tabel yang mudah dibaca dengan kolom untuk total memori, memori yang digunakan, memori yang bebas, memori yang dibagikan, cache buffer, dan memori yang benar-benar tersedia. Sedangkan jika melihat isi dari file `/proc/meminfo` formatnya berupa teks mentah dengan daftar kunci-nilai yang menunjukkan berbagai metrik penggunaan memori.

4. Gunakan perintah dibawah ini

```
$ ls -lR /.
```

```

~
> ls -lR /.
./.:
total 1428
lrwxrwxrwx 1 root root 7 Nov 23 04:36 bin → usr/bin
drwxr-xr-x 2 root root 4096 Apr 18 2022 boot
drwxr-xr-x 8 root root 2760 May 2 19:45 dev
drwxr-xr-x 96 root root 4096 May 2 19:45 etc
drwxr-xr-x 3 root root 4096 Feb 24 16:06 home
-rwxr-xr-x 2 root root 1392816 Feb 22 11:13 init
lrwxrwxrwx 1 root root 7 Nov 23 04:36 lib → usr/lib
lrwxrwxrwx 1 root root 9 Nov 23 04:36 lib32 → usr/lib32
lrwxrwxrwx 1 root root 9 Nov 23 04:36 lib64 → usr/lib64
lrwxrwxrwx 1 root root 10 Nov 23 04:36 libx32 → usr/libx32
drwx----- 2 root root 16384 Apr 10 2019 lost+found
drwxr-xr-x 2 root root 4096 Nov 23 04:36 media
drwxr-xr-x 5 root root 4096 Feb 24 16:05 mnt
drwxr-xr-x 2 root root 4096 Nov 23 04:36 opt
dr-xr-xr-x 149 root root 0 May 2 19:45 proc
drwx----- 3 root root 4096 Mar 5 20:18 root
drwxr-xr-x 6 root root 120 May 2 19:45 run
lrwxrwxrwx 1 root root 8 Nov 23 04:36 sbin → usr/sbin
drwxr-xr-x 8 root root 4096 Nov 23 04:37 snap
drwxr-xr-x 2 root root 4096 Nov 23 04:36 srv
dr-xr-xr-x 11 root root 0 May 2 19:45 sys
drwxrwxrwt 3 root root 4096 May 2 19:45 tmp
drwxr-xr-x 14 root root 4096 Nov 23 04:36 usr
drwxr-xr-x 13 root root 4096 Nov 23 04:37 var

```

Menampilkan daftar detail dari semua file dan direktori yang ada di root.

- Perhatikan perubahan manajemen memory

\$ free

```

~
> free

```

	total	used	free	shared	buff/cache	available
Mem:	3921212	67204	3798712	64	55296	3710396
Swap:	1048576	0	1048576			

Jumlah memori yang terpakai (used) bertambah dari yang sebelumnya 65852 menjadi 67204.

- Jalankan sebuah program, misalnya open Office. Perhatikan perubahan manajemen memory.

\$ free

```

~
> python3
Python 3.10.12 (main, Nov 20 2023, 15:14:05) [GCC 11.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> print ("Hello World !")
Hello World !
>>>

```

Disini saya menjalankan program python.

```

~
> free

```

	total	used	free	shared	buff/cache	available
Mem:	3921212	67252	3792448	64	61512	3707248
Swap:	1048576	0	1048576			

Jumlah memori yang terpakai (used) bertambah.

7. Dengan perintah ps bagaimana penggunaan memory untuk setiap proses diatas ?

\$ ps -uax

```

~
> ps -uax

```

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	0.0	0.0	1768	1092	?	Sl	09:25	0:00	/init
root	13	0.0	0.0	2128	356	?	Ss	09:25	0:00	/init
root	14	0.0	0.0	2128	368	?	S	09:25	0:00	/init
lavina	15	0.0	0.2	10460	8764	pts/0	Ss	09:25	0:00	-zsh
lavina	57	0.0	0.1	9876	5972	pts/0	S	09:25	0:00	-zsh
lavina	62	0.0	0.0	3452	1084	pts/0	Sl	09:25	0:00	/home/lavina/.cache/gitstatus/gitstatusd-linux-x86_64 -
lavina	143	0.0	0.0	7788	3356	pts/0	R+	09:36	0:00	ps -uax

Setelah melakukan percobaan diatas disini terlihat jika memori yang digunakan sebesar 0.0 %.

Latihan

1. Ubahlah program fork5.cpp pada percobaan 5 untuk mengeksekusi perintah yang ekuivalen dengan
 - a. ls -al /etc.

```

~/projek/p11-so
> ls -al /etc

```

total	868
drwxr-xr-x	97 root root 4096 May 4 19:51 .
drwxr-xr-x	19 root root 4096 May 4 19:46 ..
-rw-----	1 root root 0 Nov 23 04:36 .pwd.lock
drwxr-xr-x	2 root root 4096 Nov 23 04:37 PackageKit
drwxr-xr-x	7 root root 4096 Mar 5 17:24 X11
-rw-r--r--	1 root root 3028 Nov 23 04:36 adduser.conf
-rw-r--r--	1 root root 51 Mar 6 13:52 aliases
-rw-r--r--	1 root root 12288 Mar 6 13:52 aliases.db
drwxr-xr-x	2 root root 4096 May 4 19:51 alternatives
drwxr-xr-x	3 root root 4096 Mar 5 18:47 apache2
drwxr-xr-x	3 root root 4096 Nov 23 04:37 apparmor
drwxr-xr-x	8 root root 4096 Nov 23 04:37 apparmor.d
drwxr-xr-x	3 root root 4096 Nov 23 04:37 apport
drwxr-xr-x	8 root root 4096 Nov 23 04:37 apt
-rw-r--r--	1 root root 2319 Jan 6 2022 bash.bashrc
-rw-r--r--	1 root root 45 Nov 11 2021 bash_completion
drwxr-xr-x	2 root root 4096 Nov 23 04:37 bash_completion.d
-rw-r--r--	1 root root 367 Dec 16 2020 bindresvport.blacklist
drwxr-xr-x	2 root root 4096 Apr 8 2022 binfmt.d
drwxr-xr-x	2 root root 4096 Nov 23 04:37 byobu
drwxr-xr-x	3 root root 4096 Nov 23 04:36 ca-certificates
-rw-r--r--	1 root root 6253 Nov 23 04:36 ca-certificates.conf
-rw-r--r--	1 root root 5529 Nov 23 04:36 ca-certificates.conf.dpkg-old
drwxr-xr-x	2 root root 4096 Nov 23 04:36 console-setup
drwxr-xr-x	2 root root 4096 Nov 23 04:36 cron.d
drwxr-xr-x	2 root root 4096 Mar 14 20:09 cron.daily
drwxr-xr-x	2 root root 4096 Nov 23 04:36 cron.hourly
drwxr-xr-x	2 root root 4096 Nov 23 04:36 cron.monthly
drwxr-xr-x	2 root root 4096 Nov 23 04:37 cron.weekly

b. cat fork2

```
~/projek/p11-so
> cat fork.cpp
#include <iostream>
using namespace std;
#include <sys/types.h>
#include <unistd.h>
#include <sys/wait.h>

int main(void) {
    pid_t child_pid;
    int status;
    pid_t wait_result;

    child_pid = fork();
    if (child_pid == 0)
    {
        /* code */
        cout << "I am a child and my pid = " << getpid() << endl;
        execl("/bin/ls", "ls", "-l", "/home", NULL);
        cout << "Could not execl file /bin/ls" << endl;
        exit(1);
    } else if (child_pid > 0)
    {
        /* code */
        cout << "I am the parent and my pid = " << getpid() << endl;
        cout << "My child has pid = " << child_pid << endl;
    }
    else
    {
        /* code */
        cout << "The fork system call failed to create a new process" << endl;
        exit(1);
    }
}
```

c. ./fork2

```
~/projek/p11-so
> ./fork5
I am the parent and my pid = 4379
My child has pid = 4380
I am a happy, healthy process and my pid = 4379
I am a parent and i am going to wait for my child
I am a child and my pid = 4380
total 4
drwxr-x--- 15 lavina lavina 4096 May  4 20:20 lavina
I am parent and i am quitting.
```

2. Informasi apa saja mengenai manajemen memory yang ditampilkan pada perintah dmesg pada percobaan Anda ?

Jawab : Terdapat informasi pesan kernel ring buffer. Pesan ini berisi informasi tentang proses boot sistem, perangkat keras yang dideteksi, dan aktivitas kernel lainnya. jumlah total memori, memori yang terpakai, memori yang tidak terpakai.

3. Bagaimana informasi yang ditampilkan dengan perintah free pada percobaan Anda ?

Jawab : Menampilkan total memori (total), total memori yang terpakai (used), total memori yang tidak terpakai (free), informasi buffer dan cache serta informasi tentang penggunaan ruang swap.

```
~/projek/p11-so 20:20:52
> free
```

	total	used	free	shared	buff/cache	available
Mem:	3921212	514036	1952492	68	1454684	3172292
Swap:	1048576	0	1048576			

4. Apa isi file /proc/meminfo pada percobaan yang Anda lakukan ?

```
/proc
> cat meminfo
```

MemTotal:	3921212 kB
MemFree:	1955032 kB
MemAvailable:	3176112 kB
Buffers:	37748 kB
Cached:	1343524 kB
SwapCached:	0 kB
Active:	366196 kB
Inactive:	1437136 kB
Active(anon):	196 kB
Inactive(anon):	419040 kB
Active(file):	366000 kB
Inactive(file):	1018096 kB
Unevictable:	0 kB
Mlocked:	0 kB
SwapTotal:	1048576 kB
SwapFree:	1048576 kB
Dirty:	4 kB
Writeback:	0 kB
AnonPages:	417152 kB
Mapped:	101504 kB
Shmem:	68 kB
KReclaimable:	74768 kB
Slab:	100300 kB
SReclaimable:	74768 kB
SUnreclaim:	25532 kB
KernelStack:	3360 kB
PageTables:	8528 kB
NFS_Unstable:	0 kB
Bounce:	0 kB

Jika diperhatikan file ini berisi informasi mengenai memori dengan format yang lebih lengkap.

5. Berapa besar memory yang digunakan setelah percobaan 7 dengan perintah ps – uax?

```
~ 19:52:18
> ps -uax
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0   1768  1100 ?        SL    19:45   0:00 /init
root       13  0.0  0.0    2128    364 ?        Ss    19:45   0:00 /init
root       14  0.0  0.0    2128    376 ?        S     19:45   0:00 /init
lavina    15  0.1  0.2  10336  8792 pts/0    Ss    19:45   0:00 -zsh
lavina    19  0.0  0.1   8372  4604 pts/0    S     19:45   0:00 -zsh
lavina    52  0.0  0.0   3452  1068 pts/0    SL    19:45   0:00 /home/lavina/.cache/gitstatus/gitstatusd-linux-x86_64 -G
lavina   192  0.0  0.0   7784  3404 pts/0    R+    19:52   0:00 ps -uax
```

Memori yang digunakan setelah command ps-uax adalah 0.0

6. Lakukan hal yang sama dengan percobaan 7 untuk melihat perubahan memory setelah dilakukan beberapa proses pada shell. Tentukan perintah yang dilakukan misalnya membuka browser dan perhatikan hal-hal berikut :

- a. Informasi apa saja yang ditampilkan dengan perintah free ?

```
~
> free
              total        used        free      shared  buff/cache   available
Mem:           3921212        66776      3792344           64        62092      3707432
Swap:          1048576             0          1048576
```

Berisi informasi total memori fisik yang tersedia di sistem (3921212 kB), jumlah memori yang telah digunakan (6676 kB), jumlah memori yang free (3792344 kB), jumlah memori yang digunakan di cache buffer (62092 kB) dan juga jumlah memori yang benar-benar tersedia (3707432 kB). Ada juga informasi swap yang menampung memori virtual.

- b. Informasi apa saja yang disimpan file /proc/meminfo ?

Jawab : File ini menyediakan informasi yang lebih detail tentang memori fisik, cache buffer, swap space, dan penggunaan memori oleh kernel. Lebih detailnya file ini berisi :

- MemTotal: Menunjukkan total memori fisik yang terpasang di sistem.
- MemFree: Menunjukkan memori fisik yang benar-benar bebas dan tersedia untuk digunakan.
- Buffers: Menunjukkan memori fisik yang digunakan untuk cache buffer kernel.
- Cached: Menunjukkan memori fisik yang digunakan untuk cache page kernel.
- SwapTotal: Menunjukkan total swap space yang tersedia di sistem.
- SwapFree: Menunjukkan swap space yang benar-benar bebas dan tersedia untuk digunakan.

- Slab: Menunjukkan memori fisik yang digunakan untuk cache slab kernel.
- Commit_AS: Menunjukkan jumlah memori yang dikomit oleh semua proses.

c. Berapa besar kapasitas memory total ?

Jawab : 3921212 kB

d. Berapa kapasitas memory yang sudah terpakai ?

Jawab : 6676 kB

e. Berapa kapasitas memory yang belum terpakai ?

Jawab : 3792344 kB

f. Berapa kapasitas memory yang digunakan sharing beberapa proses ?

Jawab : 64 kB

g. Berapa kapasitas buffer cache ?

Jawab : 62092 kB