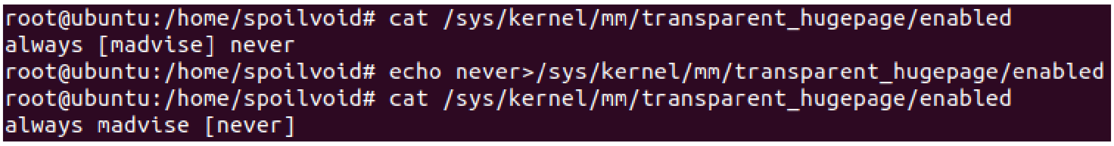
课程大作业四

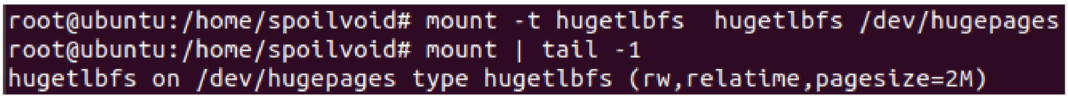
叶增渝 519030910168

1.在 VMWare 开的 Ubuntu 虚拟机中关闭 transparent_hugepage



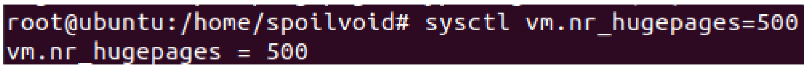
```
root@ubuntu:/home/spoilvoid# cat /sys/kernel/mm/transparent_hugepage/enabled
always [madvise] never
root@ubuntu:/home/spoilvoid# echo never>/sys/kernel/mm/transparent_hugepage/enabled
root@ubuntu:/home/spoilvoid# cat /sys/kernel/mm/transparent_hugepage/enabled
always madvise [never]
```

2.由于本虚拟机支持 hugepage 机制，所以已经存在本地目录，在这里进行挂载



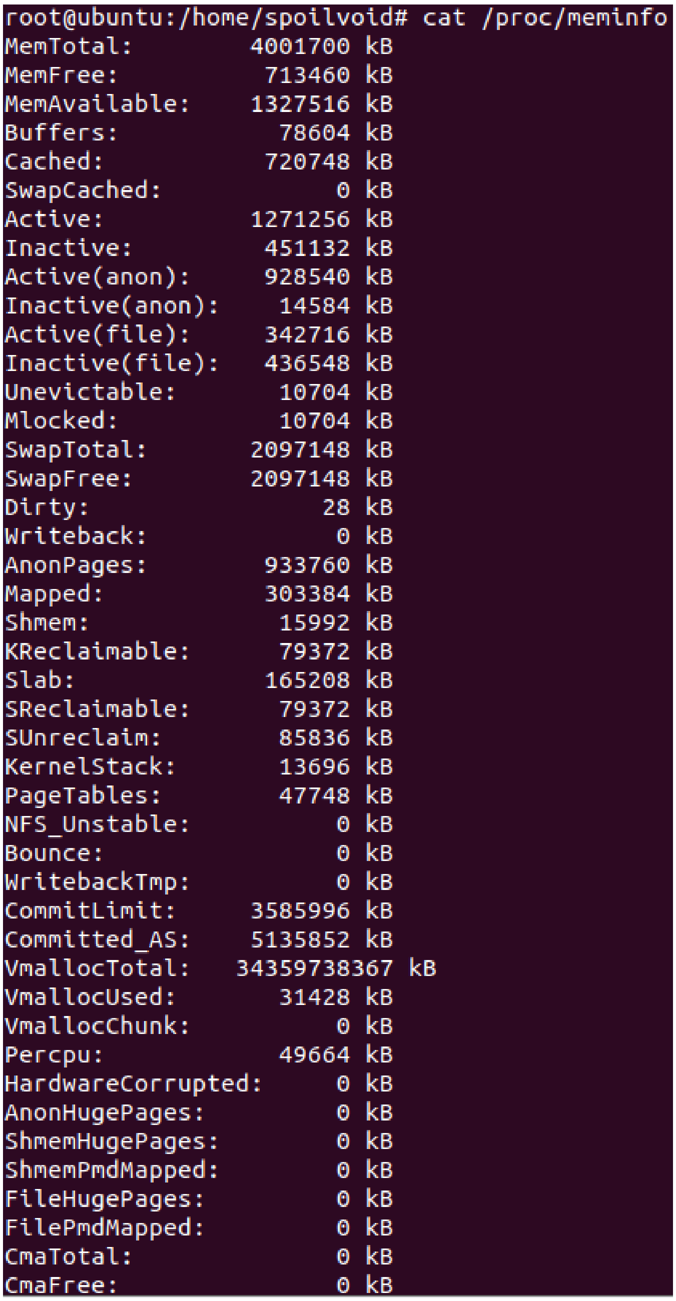
```
root@ubuntu:/home/spoilvoid# mount -t hugetlbfs  hugetlbfs /dev/hugepages
root@ubuntu:/home/spoilvoid# mount | tail -1
hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime,pagesize=2M)
```

可以看到 hugepage 的 TLB 表已经链接到对应的位置上，每个 hugepage 的大小为 2M
3.设置 hugepage 的数量为 500 个，即总共 1G 的 huagepage

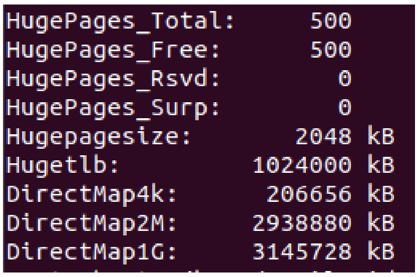


```
root@ubuntu:/home/spoilvoid# sysctl vm.nr_hugepages=500
vm.nr_hugepages = 500
```

4.查看当前的 mem 配置文件

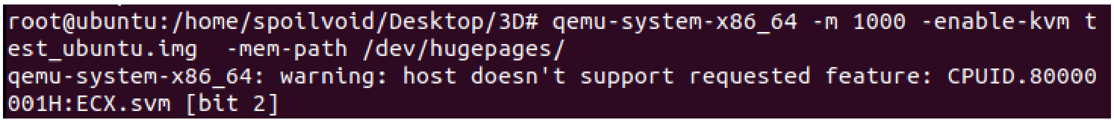


```
root@ubuntu:/home/spoilvoid# cat /proc/meminfo
MemTotal:        4001700 kB
MemFree:          713460 kB
MemAvailable:    1327516 kB
Buffers:           78604 kB
Cached:           720748 kB
SwapCached:            0 kB
Active:          1271256 kB
Inactive:         451132 kB
Active(anon):     928540 kB
Inactive(anon):    14584 kB
Active(file):     342716 kB
Inactive(file):   436548 kB
Unevictable:       10704 kB
Mlocked:           10704 kB
SwapTotal:       2097148 kB
SwapFree:        2097148 kB
Dirty:                28 kB
Writeback:             0 kB
AnonPages:        933760 kB
Mapped:           303384 kB
Shmem:             15992 kB
KReclaimable:      79372 kB
Slab:             165208 kB
SReclaimable:      79372 kB
SUnreclaim:        85836 kB
KernelStack:       13696 kB
PageTables:        47748 kB
NFS_Unstable:          0 kB
Bounce:                0 kB
WritebackTmp:          0 kB
CommitLimit:     3585996 kB
Committed_AS:    5135852 kB
VmallocTotal:   34359738367 kB
VmallocUsed:       31428 kB
VmallocChunk:          0 kB
Percpu:            49664 kB
HardwareCorrupted:     0 kB
AnonHugePages:         0 kB
ShmemHugePages:        0 kB
ShmemPmdMapped:        0 kB
FileHugePages:         0 kB
FilePmdMapped:         0 kB
CmaTotal:              0 kB
CmaFree:               0 kB
```



```
HugePages_Total:     500
HugePages_Free:      500
HugePages_Rsvd:        0
HugePages_Surp:        0
Hugepagesize:       2048 kB
Hugetlb:         1024000 kB
DirectMap4k:      206656 kB
DirectMap2M:     2938880 kB
DirectMap1G:     3145728 kB
```

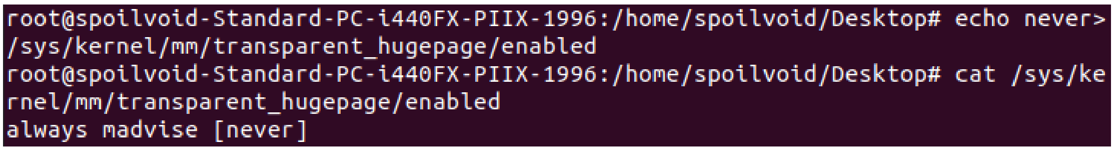
4.我们分配与 hugepage 大小相同的内存，并且将内存位置指向我们创建 hugepage 的目录，即 host 机器 allocate hugepage



```
root@ubuntu:/home/spoilvoid/Desktop/3D# qemu-system-x86_64 -m 1000 -enable-kvm t
est_ubuntu.img  -mem-path /dev/hugepages/
qemu-system-x86_64: warning: host doesn't support requested feature: CPUID.80000
001H:ECX.svm [bit 2]
```

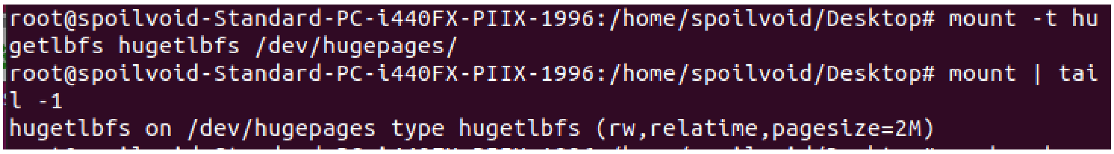
5.在打开的 QEMU 虚拟机上下载 sysbench 测试工具，并如上配置 hugepage

将 transparent_hugepage 关闭



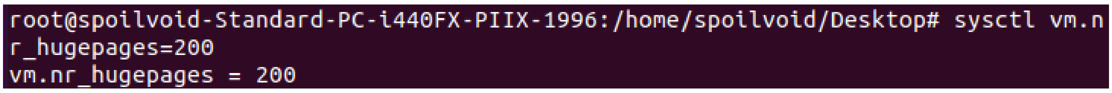
```
root@spoilvoid-Standard-PC-i440FX-PIIX-1996:/home/spoilvoid/Desktop# echo never>
/sys/kernel/mm/transparent_hugepage/enabled
root@spoilvoid-Standard-PC-i440FX-PIIX-1996:/home/spoilvoid/Desktop# cat /sys/ke
rnel/mm/transparent_hugepage/enabled
always madvise [never]
```

挂载 hugepage 目录.每个 hugepage 大小为 2M



```
root@spoilvoid-Standard-PC-i440FX-PIIX-1996:/home/spoilvoid/Desktop# mount -t hu
getlbfs hugetlbfs /dev/hugepages/
root@spoilvoid-Standard-PC-i440FX-PIIX-1996:/home/spoilvoid/Desktop# mount | tai
l -1
hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime,pagesize=2M)
```

设置 hugepage 数量为 200

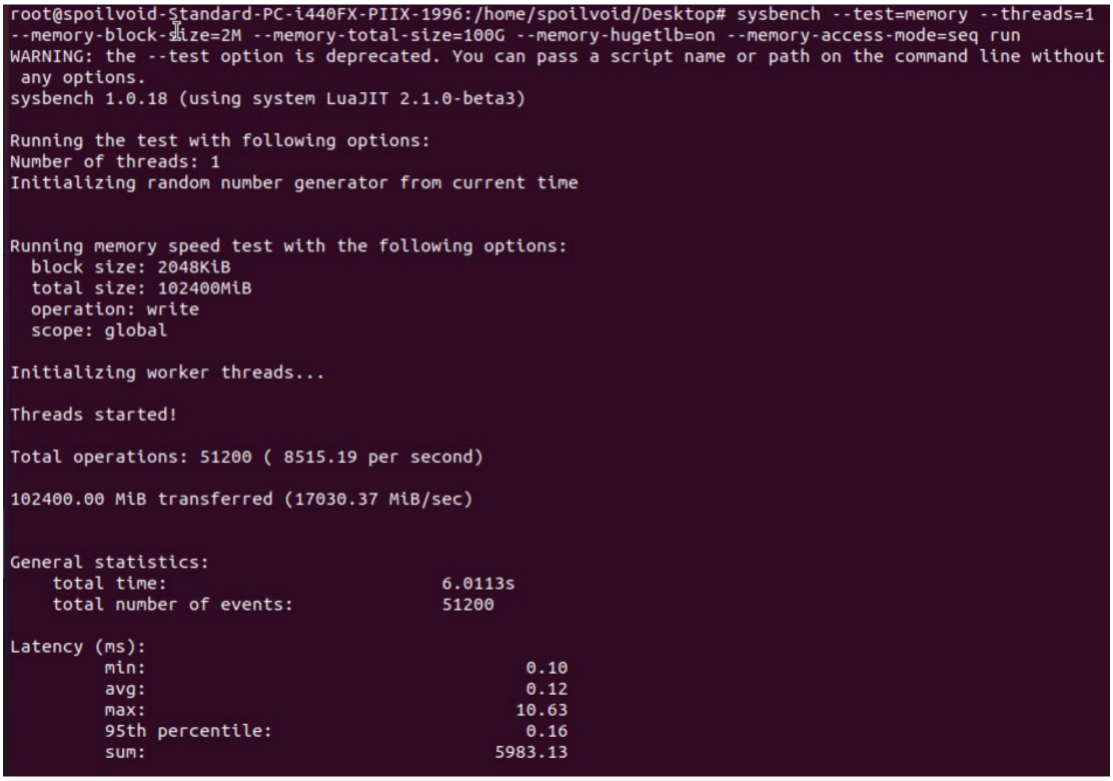


```
root@spoilvoid-Standard-PC-i440FX-PIIX-1996:/home/spoilvoid/Desktop# sysctl vm.n
r_hugepages=200
vm.nr_hugepages = 200
```

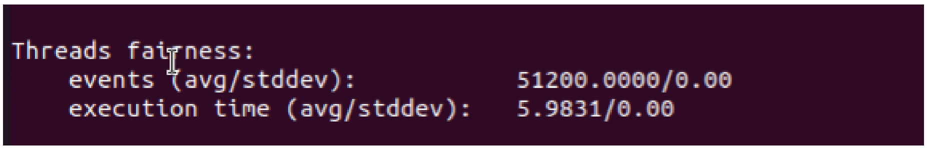
6.在 host 机 allocate hugepage 的情况下在 QEMU 虚拟机内 use hugepage 进行 sysbench memory test

(1)host 机 allocate hugepage，QEMU use hugepage：

下方命令的含义为进行内存测试，线程数为 1，每一个 block 为 2M 大小，总测试数据量为 100G，从 hugetlb 即之前 hugepage 挂载的目录分配内存，进行顺序存储



```
root@spoilvoid-Standard-PC-i440FX-PIIX-1996:/home/spoilvoid/Desktop# sysbench --test=memory --threads=1
--memory-block-size=2M --memory-total-size=100G --memory-hugetlb=on --memory-access-mode=seq run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without
 any options.
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time


Running memory speed test with the following options:
  block size: 2048KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 51200 ( 8515.19 per second)

102400.00 MiB transferred (17030.37 MiB/sec)


General statistics:
    total time:                          6.0113s
    total number of events:              51200

Latency (ms):
         min:                                    0.10
         avg:                                    0.12
         max:                                   10.63
         95th percentile:                        0.16
         sum:                                 5983.13
```

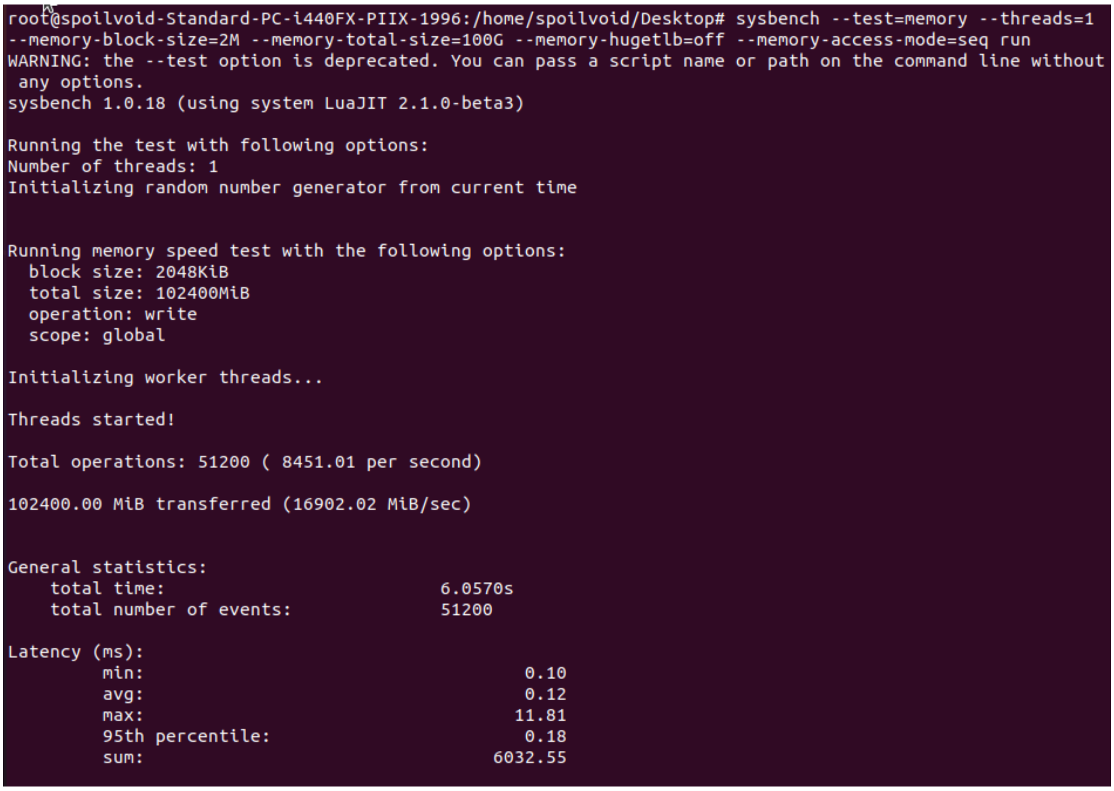


```
Threads fairness:
    events (avg/stddev):           51200.0000/0.00
    execution time (avg/stddev):   5.9831/0.00
```

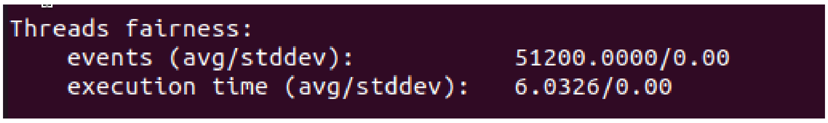
最终得到 transfer rate 为 17030.37MiB/sec

(2) host 机 allocate hugepage，QEMU not use hugepage：

下方命令的含义为进行内存测试，线程数为 1，每一个 block 为 2M 大小，总测试数据量为 100G，不从 hugetlb 中分配内存，进行顺序存储



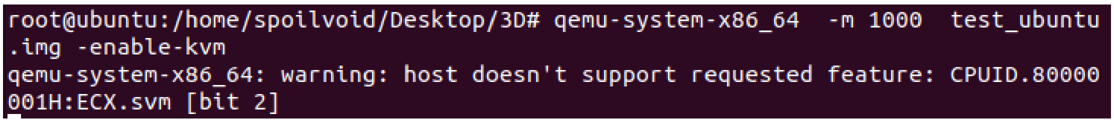
```
root@spoilvoid-Standard-PC-i440FX-PIIX-1996:/home/spoilvoid/Desktop# sysbench --test=memory --threads=1
--memory-block-size=2M --memory-total-size=100G --memory-hugetlb=off --memory-access-mode=seq run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without
 any options.
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time


Running memory speed test with the following options:
  block size: 2048KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 51200 ( 8451.01 per second)

102400.00 MiB transferred (16902.02 MiB/sec)


General statistics:
    total time:                          6.0570s
    total number of events:              51200

Latency (ms):
         min:                                    0.10
         avg:                                    0.12
         max:                                   11.81
         95th percentile:                        0.18
         sum:                                 6032.55
```



```
Threads fairness:
    events (avg/stddev):           51200.0000/0.00
    execution time (avg/stddev):   6.0326/0.00
```

最终得到 transfer rate 为 16902.02MiB/sec

7. host 机同样分配大小相同的 1000M 内存，不使用 huagepage 直接打开 QEMU 虚拟机，即 host not allocate hugepage

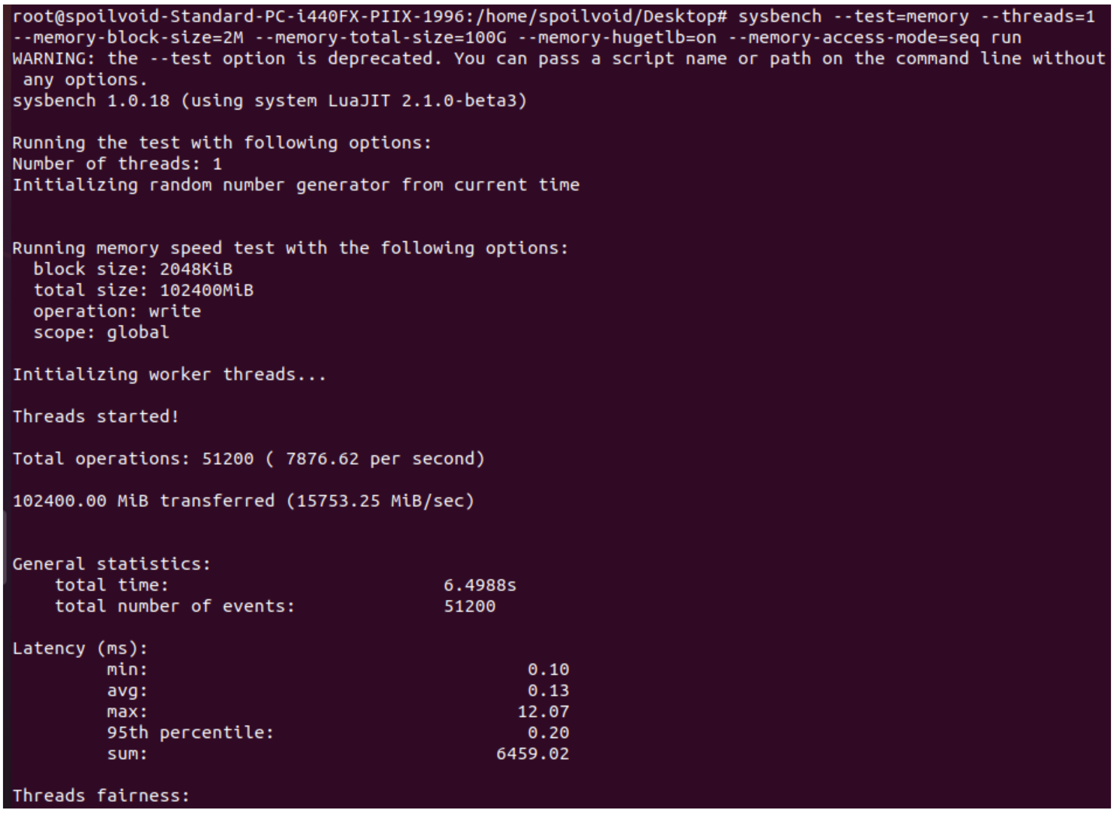


```
root@ubuntu:/home/spoilvoid/Desktop/3D# qemu-system-x86_64  -m 1000  test_ubuntu
.img -enable-kvm
qemu-system-x86_64: warning: host doesn't support requested feature: CPUID.80000
001H:ECX.svm [bit 2]
```

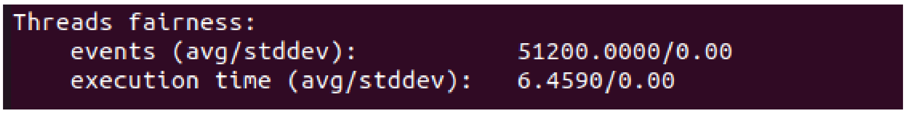
如上第 5 步配置虚拟机 hugepage 并关闭 transparent_hugepage 分配 200 个 2M 大小的 hugepage

(1)host 机 not allocate hugepage，QEMU use hugepage：

下方命令的含义为进行内存测试，线程数为 1，每一个 block 为 2M 大小，总测试数据量为 100G，从 hugetlb 即之前 hugepage 挂载的目录分配内存，进行顺序存储



```
root@spoilvoid-Standard-PC-i440FX-PIIX-1996:/home/spoilvoid/Desktop# sysbench --test=memory --threads=1
--memory-block-size=2M --memory-total-size=100G --memory-hugetlb=on --memory-access-mode=seq run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without
 any options.
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time


Running memory speed test with the following options:
  block size: 2048KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 51200 ( 7876.62 per second)

102400.00 MiB transferred (15753.25 MiB/sec)


General statistics:
    total time:                          6.4988s
    total number of events:              51200

Latency (ms):
         min:                                    0.10
         avg:                                    0.13
         max:                                   12.07
         95th percentile:                        0.20
         sum:                                 6459.02

Threads fairness:
```

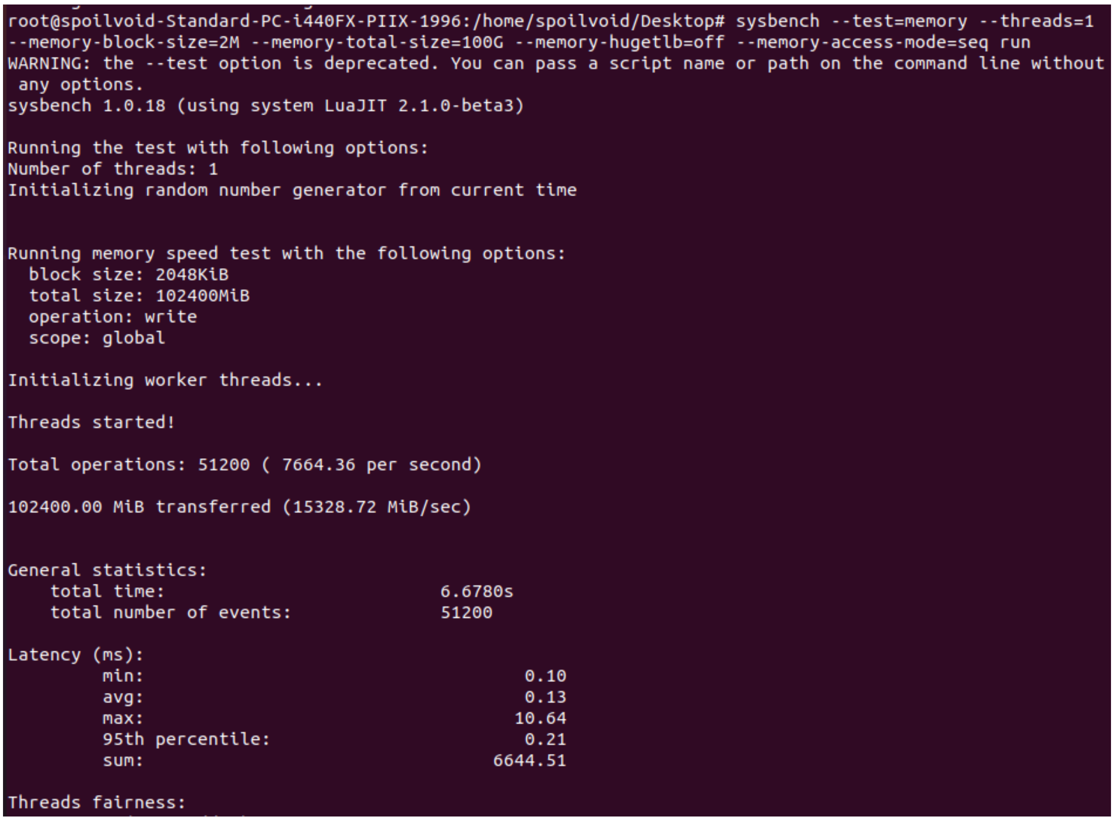


```
Threads fairness:
    events (avg/stddev):           51200.0000/0.00
    execution time (avg/stddev):   6.4590/0.00
```

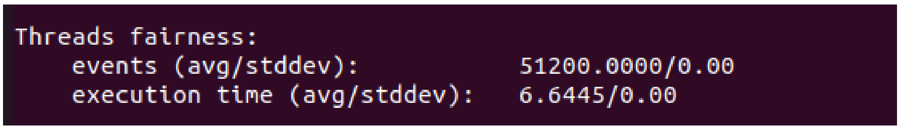
最终得到 transfer rate 为 15735.25MiB/sec

(2) host 机 not allocate hugepage，QEMU not use hugepage：

下方命令的含义为进行内存测试，线程数为 1，每一个 block 为 2M 大小，总测试数据量为 100G，不从 hugetlb 中分配内存，进行顺序存储



```
root@spoilvoid-Standard-PC-i440FX-PIIX-1996:/home/spoilvoid/Desktop# sysbench --test=memory --threads=1
--memory-block-size=2M --memory-total-size=100G --memory-hugetlb=off --memory-access-mode=seq run
WARNING: the --test option is deprecated. You can pass a script name or path on the command line without
 any options.
sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time


Running memory speed test with the following options:
  block size: 2048KiB
  total size: 102400MiB
  operation: write
  scope: global

Initializing worker threads...

Threads started!

Total operations: 51200 ( 7664.36 per second)

102400.00 MiB transferred (15328.72 MiB/sec)


General statistics:
    total time:                          6.6780s
    total number of events:              51200

Latency (ms):
         min:                                    0.10
         avg:                                    0.13
         max:                                   10.64
         95th percentile:                        0.21
         sum:                                 6644.51

Threads fairness:
```



```
Threads fairness:
    events (avg/stddev):           51200.0000/0.00
    execution time (avg/stddev):   6.6445/0.00
```

最终得到 transfer rate 为 15328.72MiB/sec

8.实验结果总结

Transfer rate	Host allocate hugepage	Host not allocate hugepage
QEMU use hugepage	17030.37MiB/sec	15735.25MiB/sec
QEMU not use hugepage	17705.43MiB/sec	15328.72 MiB/sec

可以看到在 host 机 allocate hugepage 的时候，相比起不 allocate hugepage，transfer rate 有较大提升，在 QEMU 虚拟机中使用 hugepage 确实能提高一定 transfer rate，但是效果不是很明显。

可能的解释：使用 hugepage 使得 TLB 表项减少，在查询真实地址时的时间减少，从而提升了 transfer rate，而在 QEMU 虚拟机内由于本身由 host 机分配内存小，所以造成区别不大