CPSC 457 WINTER 2021
30009135
Brandon Nguyen

Q1.

a)

slow-pali.cpp palindrome.py

t3.txt

```
real    0m0.006s             0m0.029s
user    0m0.002s            0m0.020s
sys     0m0.003s             0m0.007s
```

t4.txt

```
real    0m2.856s             0m0.304s
user    0m1.405s             0m0.292s
sys     0m1.447s             0m0.009s
```

b)

For t3.txt the *C++* program **user time** was 0.002s, while the **user time** for the *Python* program was 0.020s. The **kernel time** for the *C++* program was 0.003s, while the **kernel time** for the *Python* program was 0.007s.

For t4.txt the *C++* program **user time** was 1.405s, while the **user time** for the *Python* program was 0.292s. The **kernel time** for the *C++* program was 1.447s, while the **kernel time** for the *Python* program was 0.009s.

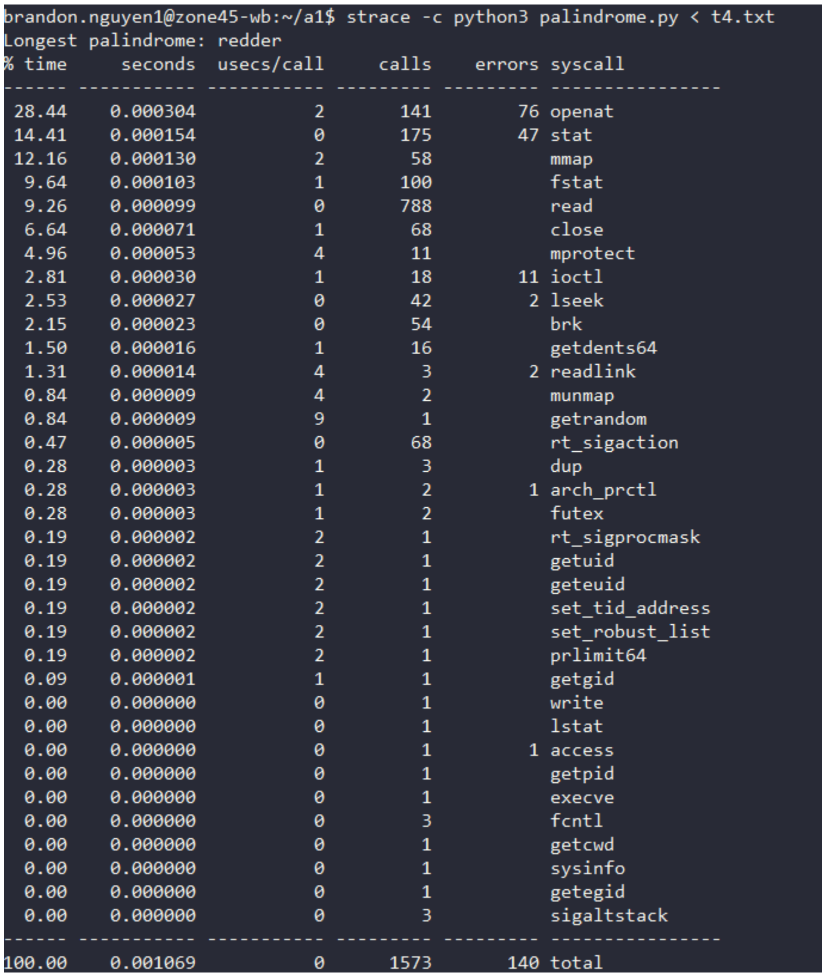
c)

For t3.txt the *C++* program uses less system calls than the *Python* version, which makes the *C++* version of the program the fastest. Most of the time in *Python* version was spent on the CPU.

For t4.txt the *C++* program calls the read system call a numerous number of times more than the *Python* version, making it slower. While the *Python* program uses more system calls but, uses less calls on each function, which decreases the time spent in kernel mode.

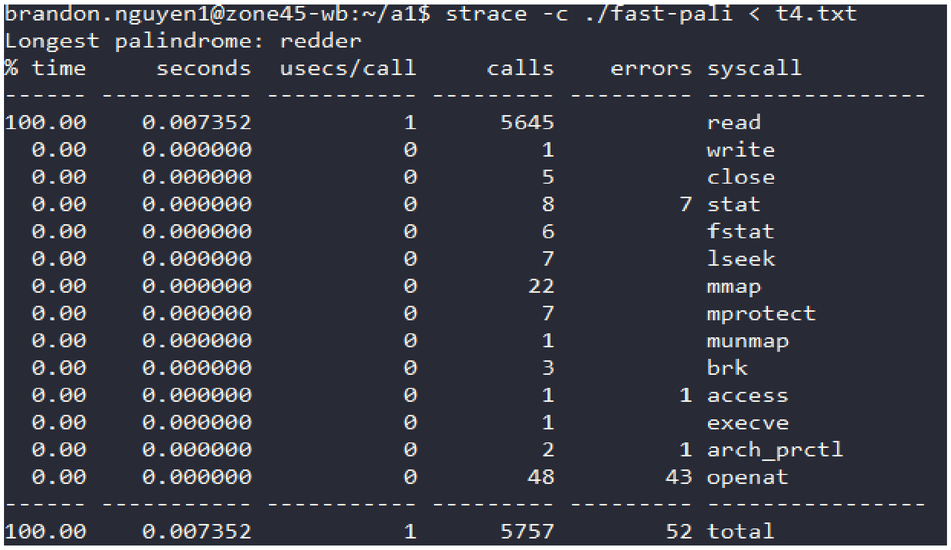
Q3.

palindrom.py when reading t4.txt



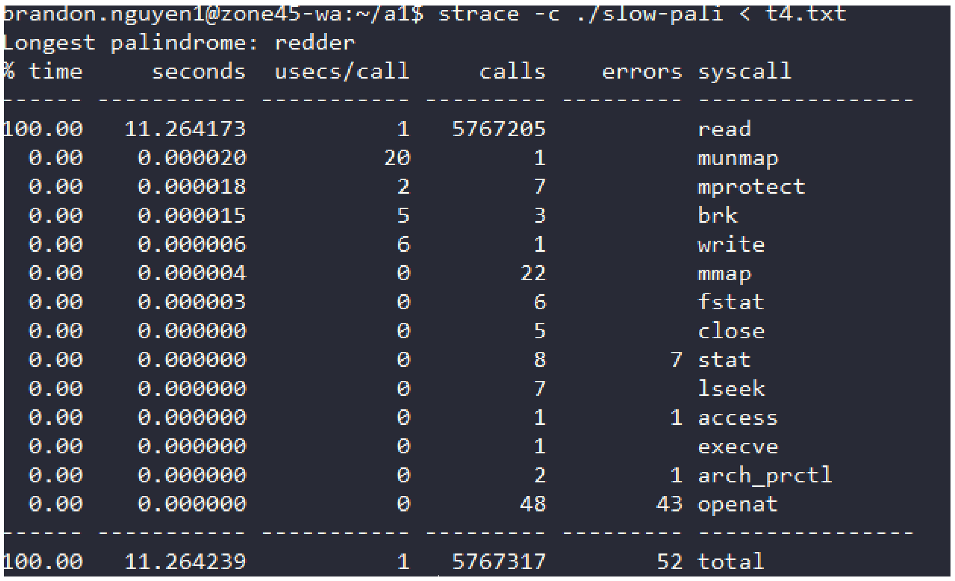
```
brandon.nguyen1@zone45-wb:~/a1$ strace -c python3 palindrome.py < t4.txt
Longest palindrome: redder
% time     seconds  usecs/call     calls    errors syscall
------ ----------- ----------- --------- --------- ----------------
 28.44    0.000304           2       141        76 openat
 14.41    0.000154           0       175        47 stat
 12.16    0.000130           2        58           mmap
  9.64    0.000103           1       100           fstat
  9.26    0.000099           0       788           read
  6.64    0.000071           1        68           close
  4.96    0.000053           4        11           mprotect
  2.81    0.000030           1        18        11 ioctl
  2.53    0.000027           0        42         2 lseek
  2.15    0.000023           0        54           brk
  1.50    0.000016           1        16           getdents64
  1.31    0.000014           4         3         2 readlink
  0.84    0.000009           4         2           munmap
  0.84    0.000009           9         1           getrandom
  0.47    0.000005           0        68           rt_sigaction
  0.28    0.000003           1         3           dup
  0.28    0.000003           1         2         1 arch_prctl
  0.28    0.000003           1         2           futex
  0.19    0.000002           2         1           rt_sigprocmask
  0.19    0.000002           2         1           getuid
  0.19    0.000002           2         1           geteuid
  0.19    0.000002           2         1           set_tid_address
  0.19    0.000002           2         1           set_robust_list
  0.19    0.000002           2         1           prlimit64
  0.09    0.000001           1         1           getgid
  0.00    0.000000           0         1           write
  0.00    0.000000           0         1           lstat
  0.00    0.000000           0         1         1 access
  0.00    0.000000           0         1           getpid
  0.00    0.000000           0         1           execve
  0.00    0.000000           0         3           fcntl
  0.00    0.000000           0         1           getcwd
  0.00    0.000000           0         1           sysinfo
  0.00    0.000000           0         1           getegid
  0.00    0.000000           0         3           sigaltstack
------ ----------- ----------- --------- --------- ----------------
100.00    0.001069           0      1573       140 total
```

fast-pali.cpp reading in t4.txt



```
brandon.nguyen1@zone45-wb:~/a1$ strace -c ./fast-pali < t4.txt
Longest palindrome: redder
% time     seconds  usecs/call     calls    errors syscall
------ ----------- ----------- --------- --------- ----------------
100.00    0.007352           1      5645           read
  0.00    0.000000           0         1           write
  0.00    0.000000           0         5           close
  0.00    0.000000           0         8         7 stat
  0.00    0.000000           0         6           fstat
  0.00    0.000000           0         7           lseek
  0.00    0.000000           0        22           mmap
  0.00    0.000000           0         7           mprotect
  0.00    0.000000           0         1           munmap
  0.00    0.000000           0         3           brk
  0.00    0.000000           0         1         1 access
  0.00    0.000000           0         1           execve
  0.00    0.000000           0         2         1 arch_prctl
  0.00    0.000000           0        48        43 openat
------ ----------- ----------- --------- --------- ----------------
100.00    0.007352           1      5757        52 total
```

slow-pali.cpp reading in t4.txt


```
prandon.nguyen1@zone45-wa:~/a1$ strace -c ./slow-pali < t4.txt
Longest palindrome: redder
% time     seconds  usecs/call     calls    errors syscall
------ ----------- ----------- --------- --------- ----------------
100.00   11.264173           1   5767205           read
  0.00    0.000020          20         1           munmap
  0.00    0.000018           2         7           mprotect
  0.00    0.000015           5         3           brk
  0.00    0.000006           6         1           write
  0.00    0.000004           0        22           mmap
  0.00    0.000003           0         6           fstat
  0.00    0.000000           0         5           close
  0.00    0.000000           0         8         7 stat
  0.00    0.000000           0         7           lseek
  0.00    0.000000           0         1         1 access
  0.00    0.000000           0         1           execve
  0.00    0.000000           0         2         1 arch_prctl
  0.00    0.000000           0        48        43 openat
------ ----------- ----------- --------- --------- ----------------
100.00   11.264239           1   5767317        52 total
```

a) My version of *fast-pali.cpp* is significantly faster than *slow-pali.cpp*. When comparing both programs, *fast-pali.cpp* makes less use of the read system call. The reduction in the number of calls allows for *fast-pali.cpp* to spend less time in kernel mode, which makes it more optimized than *slow-pali.cpp.*

b) *Fast-pali.cpp* was also significantly faster than the *Python* version. Because of the it is more optimized than *slow-pali.cpp*, *fast-pali.cpp* used lesser number of system calls when compared to the python version. In general, C++ programs should run faster than Python programs, when executing the same input.