Students: Mansoor Ali, Illya Gordyy

1. (25 Points) Briefly describe the semantics of the following Unix system calls:
a) fork( ); - **Creates additional processes which is known as a child process that is simultaneous with the parent process.**
b) exit( ); - **It terminates a process while also returning an exit code.**
c) getpid( ); - **It returns an ID of a current process**
d) getppid( ); - **This command returns an ID of a parent of the current process.**
e) wait( ); - **Makes the calling process wait until one of its child processes terminates.**
f) execl( ); - **This command substitutes a current process with another process.**

Write a C program that uses each of the above system calls at least once. Show your program listing with comments and corresponding output. Make sure that your program compiles and executes without error.

This shows the output of the test cases for each system call on the ECS-Machin: (Code is attached on canvas.)
Stopped output shows the process of wait()
Parent and child is the output using getpid(),getppid(), and fork()
Terminating the child process uses exit().
Parent process is up. Shows that completion of the wait() and then runs a ls -la command using execl() command

```
[igordyy@ecs-pa-coding1 hw1]$ ./question1
Stopped the parent process
Parent process ID: 172984
Child process ID: 172985
Terminating the child process
Parent process is up
total 76
drwxr-xr-x. 2 igordyy domain users  4096 Jul 15 16:12 .
drwxr-xr-x. 3 igordyy domain users  4096 Jul 12 23:57 ..
-rwxr-xr-x. 1 igordyy domain users 17504 Jul 10 18:56 file
-rw-r--r--. 1 igordyy domain users    80 Jul 10 18:56 file.c
-rwxr-xr-x. 1 igordyy domain users 17856 Jul 10 19:17 prog
-rwxr-xr-x. 1 igordyy domain users 17864 Jul 15 16:12 question1
-rw-r--r--. 1 igordyy domain users   984 Jul 15 16:02 question1.c
-rw-r--r--. 1 igordyy domain users     5 Jul 15 00:18 txt.txt
[igordyy@ecs-pa-coding1 hw1]$
```

2. (25 Points) Consider the evolution of computers in general and the increase in processor speed in particular. Suppose we were to design a system with a processor that is clocked at 4GHz. Further let us suppose that signals in our system can propagate at 75% of the speed of light. Determine the maximum distance between the CPU and Memory (or Cache) if we want to access data within one processor cycle. To access data, a signal must be sent by the CPU to the memory unit and the data has to be returned. How would the distance change for a system that is clocked at 12 GHz? Show how you derived your answer!

1)
4 Ghz = 4x10^9 instruction cycle per second
Speed = 3x10^8 m/s
Time for 1 processor cycle = 1/(4x10^9) seconds at 4Ghz

Distance from CPU to the memory unit = D

Since a signal must also be returned back CPU the total distance will be equal 2D

The overall formula would be

2D = Time for 1 processor cycle * Speed* 0.75

2D = 1/(4*10^9) * 3*10^8*0.75 = 0.05625 m

D = 0.05625/2 = 0.028125 m

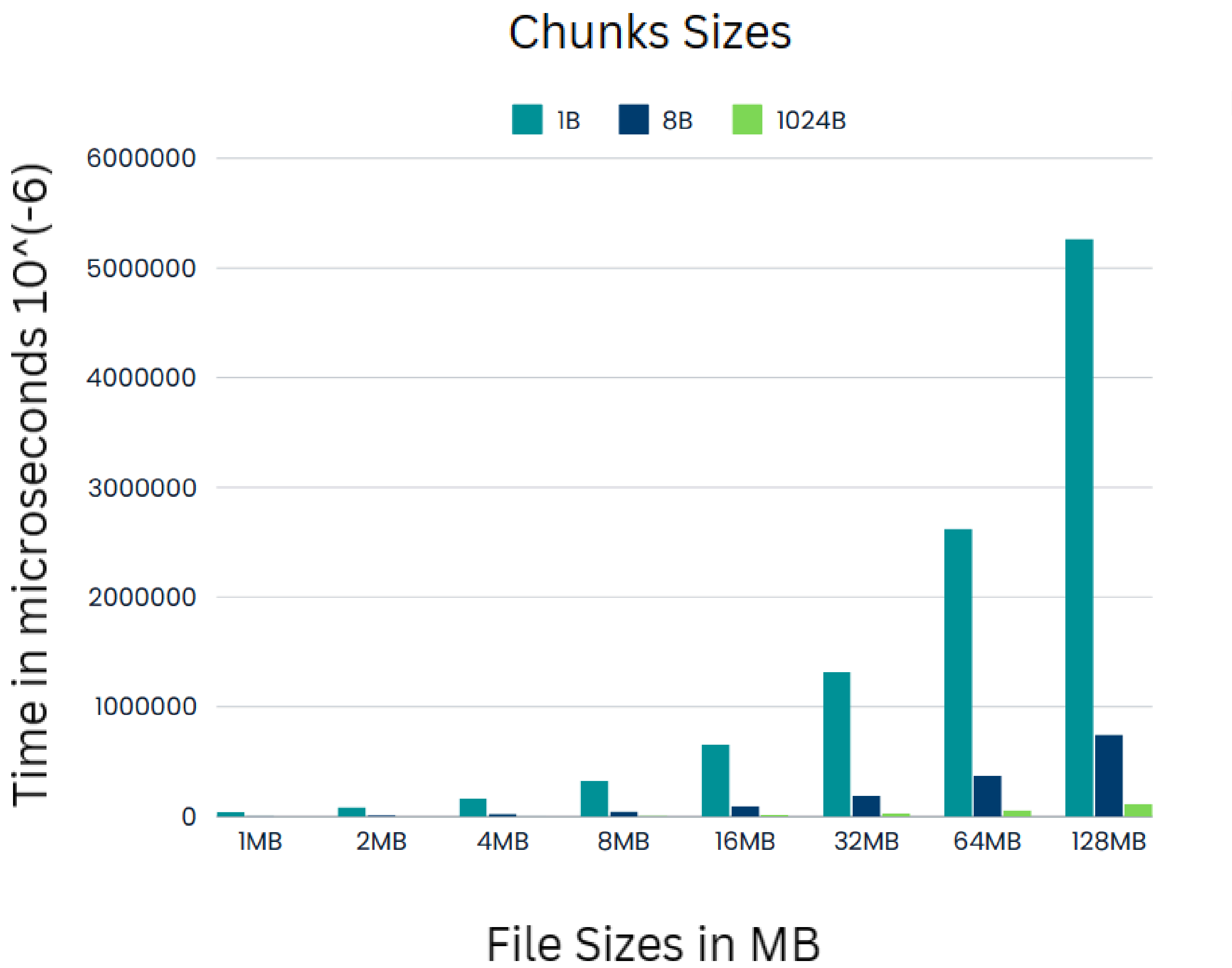
For 12 Ghz

2D = 1/(12*10^9) * 3*10^8*0.75 = 0.01875 m

D = 0.01875/2 = 0.009375 m

As the instruction cycle speed increases the distance decreases.

3. (25 Points) Write a small C program that copies data from a file A to a file B byte_by_byte. For different file sizes 1MB – 128MB (doubling the file size in each step) record the time your program requires to complete the file copy when using read() and write() system calls. Generate a graph that depicts the program performance. Repeat this experiment, but instead of copying individual bytes, use larger size portions of the file that are copied in each read() and write() system call. For instance, you may choose to copy the file in chunks of 2, 4, 8, ..., 1024 byte units. Generate a performance graph and interpret your experimental results. What do you observe? Why do you think the system shows the observed behavior? For this program it is imperative that file A and file B are located on the local disk. On Linux, the \tmp directory is located on the local file system. You must submit your program, the graphs of the corresponding experiments, and your analysis.



Our test was executed in chucks 1 byte, 8 byte, and 1024 byte. Each test consisted of these three chunks and with different file sizes ranging from 1MB-128MB. The Y-axis is milliseconds and the X-axis is the file size test cases for each chunk. Based on our observation at the initial test of each chunk of 1MB for 1b,8b, and 1024b the time to complete the file was noticeably different. Moreover, as the file size increased the buffer of size 1024 bytes demonstrated much higher performance over the other two buffer sizes. Smaller buffers resulted in much more read and write operations to read the same amount of data. Each read and write operation caused overhead, so fewer operations with larger buffers resulted in a much better performance. Moreover, the larger the buffer was, the less time it needed to allocate and deallocate memory.

```
Chunk 1 bytes:  41585  81098  162211  327333  644693  1304454  2615328  5217794
Chunk 8 bytes:  5907  11758  24089  47491  93894  188510  379628  758021
Chunk 1024 bytes:  956  1850  3659  7535  16392  31956  65134  128353
[igordyy@ecs-pa-coding1 csc139]$ ./question3
Chunk 1 bytes:  41050  81843  164309  324287  654653  1317374  2623416  5265286
Chunk 8 bytes:  5773  11696  23105  46434  92799  186250  373237  746129
Chunk 1024 bytes:  846  1696  3614  7148  14730  28489  56280  114605
```

The above example showcases the output on the ECS machine with its respective chunks and file sizes.

4. (25 Points) Imagine you are the IT manager for a growing tech company that develops high-demand software applications. The company is planning to upgrade its computing infrastructure to handle increased workloads efficiently. Your team is evaluating three types of systems: single processor systems, multiprocessor systems, and clustered systems. Given the scenario, which system (single processor, multiprocessor, or clustered) would you recommend for the company's infrastructure upgrade? Justify your recommendation with specific reasons related to performance, scalability, cost, and reliability.

We would recommend symmetric clustered systems since they offer significant advantages over single-processor and multiprocessor systems. Moreover, in case one of the nodes fails, the others continue their processing task, compared to the asymmetric one it does not keep the node on hot standby. They are much more scalable, allowing the IT department to add new machines depending on their needs, which helps companies adapt to changing requirements and decide on the cost. Furthermore, clustered systems provide high reliability due to their architecture. By storing data across multiple machines, they ensure high data availability and system functionality even if individual machines fail. Clustered systems consist of numerous interconnected machines or sets of multiple servers working together. This offers a significant performance by providing parallel processing and distributing the heavy tasks and processes on several nodes which provides faster computation of workloads.