Quiz 1 (Total Points: 100)

Your Name: _____

1. Explain the idea of limited direct execution. (5 points)

2. Explain the concept of cache affinity. (5 points)

3. **The Forked Nursery**. Examine the following C program fragment. How many **new** processes are created by the following code? (*Hint:* Draw the process tree to make this easier) (10 points)

```
/* process A */
/* ... */
fork();
fork();
fork();
```

   (a) 3 new processes
   (b) 8 new processes
   (c) **7 new processes**
   (d) 6 new processes

4. **SJF Scheduler**. Compute the average response time and average turnaround time when running three jobs with different lengths: 100,200,400, with the SJF scheduler. Assume all three jobs arrive at the same time. (15 points)

```
Execution trace:
  [ time   0 ] Run job 0 for 100.00 secs ( DONE at 100.00 )
  [ time 100 ] Run job 1 for 200.00 secs ( DONE at 300.00 )
  [ time 300 ] Run job 2 for 400.00 secs ( DONE at 700.00 )

Final statistics:
  Job   0 -- Response: 0.00  Turnaround 100.00
  Job   1 -- Response: 100.00  Turnaround 300.00
  Job   2 -- Response: 300.00  Turnaround 700.00

  Average -- Response: 133.33  Turnaround 366.67
```

5. **Round Robin Scheduler**. Compute the average response time and average turnaround time when running three jobs with different lengths: 3,6,9, with the RR scheduler and a time-slice of 1. Assume job 0 will be scheduled first, job 1 next, and then job 2. (15 points)

```
Execution trace:
  [ time   0 ] Run job   0 for 1.00 secs
  [ time   1 ] Run job   1 for 1.00 secs
  [ time   2 ] Run job   2 for 1.00 secs
  [ time   3 ] Run job   0 for 1.00 secs
  [ time   4 ] Run job   1 for 1.00 secs
  [ time   5 ] Run job   2 for 1.00 secs
  [ time   6 ] Run job   0 for 1.00 secs ( DONE at 7.00 )
  [ time   7 ] Run job   1 for 1.00 secs
  [ time   8 ] Run job   2 for 1.00 secs
  [ time   9 ] Run job   1 for 1.00 secs
  [ time  10 ] Run job   2 for 1.00 secs
  [ time  11 ] Run job   1 for 1.00 secs
  [ time  12 ] Run job   2 for 1.00 secs
  [ time  13 ] Run job   1 for 1.00 secs ( DONE at 14.00 )
  [ time  14 ] Run job   2 for 1.00 secs
  [ time  15 ] Run job   2 for 1.00 secs
  [ time  16 ] Run job   2 for 1.00 secs
  [ time  17 ] Run job   2 for 1.00 secs ( DONE at 18.00 )

Final statistics:
  Job   0 -- Response: 0.00  Turnaround 7.00
  Job   1 -- Response: 1.00  Turnaround 14.00
  Job   2 -- Response: 2.00  Turnaround 18.00

  Average -- Response: 1.00  Turnaround 13.00
```

6. Given the following workload, produce a trace and compute the cpu utilization. Assume each cpu instruction takes 1 time unit, each I/O takes 5 time units to complete. (15 points) - Will not be in the final exam.

```
Produce a trace of what would happen when you run these processes:
Process 0
  cpu
  cpu
  cpu

Process 1
  io
  io
  io

Important behaviors:
  System will switch when the current process is FINISHED or ISSUES AN IO
  After IOs, the process issuing the IO will run LATER (when it is its turn)

Answer:
Stats: Total Time 19
Stats: CPU Busy 6 (31.58%)
Stats: IO Busy  12 (63.16%)
```

7. Given the following workload, produce a trace and compute the cpu utilization. Assume each cpu instruction takes 1 time unit, each I/O takes 5 time units to complete. (15 points) - Will not be in the final exam.

```
[cs HW-CPU-Intro]$ python2 process-run.py -l 3:0,3:100
Produce a trace of what would happen when you run these processes:
Process 0
  io
  io
  io

Process 1
  cpu
  cpu
  cpu

Important behaviors:
  System will switch when the current process is FINISHED or ISSUES AN IO
  After IOs, the process issuing the IO will run LATER (when it is its turn)

Answer:
Stats: Total Time 16
Stats: CPU Busy 6 (37.50%)
Stats: IO Busy  12 (75.00%)
```

8. **Multiprocessor Scheduling**. Using the following command with the simulator multi.py, we generate a workload, which includes 3 jobs (job a,b,c), on a two-CPU system. Please calcuate how much time it takes to finish the whole workload. Assume we are using a round robin scheduler. (20 points)

```
[cs HW-Sched-MultiCPU]$ ./multi.py -n 2 -L a:4:20,b:3:5,c:2:5 -q 1 -w 2
ARG seed 0
ARG job_num 3
ARG job_list a:4:20,b:3:5,c:2:5
ARG num_cpus 2
ARG quantum 1
ARG cache_size 100

Job name:a run_time:4 working_set_size:20
Job name:b run_time:3 working_set_size:5
Job name:c run_time:2 working_set_size:5
```

Note1: cache warm up time: 2 time units. i.e., if a job runs for 2 time units, the cache on that CPU becomes warm, and then the job starts running faster. Note2: cache warm up rate: 2x. i.e., a job will execute at a 2x speed when the cache becomes warm.

Answer:

```
Scheduler central queue: ['a', 'b', 'c']

   0   a [  3] cache[   ]     b [  2] cache[   ]
-------------------------------------------------
   1   c [  1] cache[   ]     a [  2] cache[   ]
-------------------------------------------------
   2   b [  1] cache[   ]     c [  0] cache[   ]
-------------------------------------------------
   3   a [  1] cache[w  ]     b [  0] cache[ w ]
-------------------------------------------------
   4   a [  0] cache[w  ]     - [   ] cache[ w ]

Finished time 5

Per-CPU stats
  CPU 0  utilization 100.00 [ warm 20.00 ]
  CPU 1  utilization 80.00 [ warm 0.00 ]
```

Note: Cache effect doesn't exhibit in this workload/trace, but will exhibit in the final exam.