操作系统作业 3

1. What are the benefits of multi-threading? Which of the following components of program state are shared across threads in a multithreaded process?

   a. Register values

   b. Heap memory

   c. Global variables

   d. Stack memory

2. Consider the following code segment:

```
pid t pid;
pid = fork();
if (pid == 0) { /* child process */
    fork();
    thread create( . . .);
}
fork();
```

   a. How many unique processes are created?

   b. How many unique threads are created?

3. The program shown in the following figure uses Pthreads. What would be the output from the program at `LINE C` and `LINE P`?

```
#include <pthread.h>
#include <stdio.h>

int value = 0;
void *runner(void *param); /* the thread */

int main(int argc, char *argv[])
{
pid_t pid;
pthread_t tid;
pthread_attr_t attr;

   pid = fork();

   if (pid == 0) { /* child process */
      pthread_attr_init(&attr);
      pthread_create(&tid,&attr,runner,NULL);
      pthread_join(tid,NULL);
      printf("CHILD: value = %d",value); /* LINE C */
   }
   else if (pid > 0) { /* parent process */
      wait(NULL);
      printf("PARENT: value = %d",value); /* LINE P */
   }
}

void *runner(void *param) {
   value = 5;
   pthread_exit(0);
}
```

4. What are the differences between ordinary pipe and named pipe?

5. List all the requirements of the entry and exit implementation when solving the critical-section problem. Analyze whether strict alternation satisfies all the requirements.

6. What is deadlock? List the four requirements of deadlock.

7. What is semaphore? Explain the functionalities of semaphore in process synchronization.

8. Please use semaphore to provide a deadlock-free solution to address the dining philosopher problem.

9. Consider the following set of processes, with the length of the CPU burst time given in milliseconds:

Process	Burst Time	Priority
P_1	10	3
P_2	1	1
P_3	2	3
P_4	1	4
P_5	5	2

The processes are assumed to have arrived in the order P1, P2, P3, P4, P5, all at time 0.

a) Draw four Gantt charts that illustrate the execution of these processes using the following scheduling algorithms: FCFS, SJF (nonpreemptive), nonpreemptive priority (a smaller priority number implies a higher priority), and RR (quantum = 1).

b) What is the turnaround time of each process for each of the scheduling algorithms in part a?

c) What is the waiting time of each process for each of these scheduling algorithms?

d) Which of the algorithms results in the minimum average waiting time (over all processes)?

10. Which of the following scheduling algorithms could result in starvation?
    a) First-come, first-served
    b) Shortest job first
    c) Round robin
    d) Priority

11. Give an example to illustrate under what circumstances rate-monotonic scheduling is inferior to earliest-deadline-first scheduling in meeting the deadlines associated with processes?