Advanced Unix programming – Quiz9

Time limit: 12 minutes

1. (6 points) We want a thread executing the function f to wait (without polling) until $rn > 1.0$, before subtracting *1.0* from rn. You can assume that some one will signal a conditional variable *cond* whenever rn becomes larger than 1.0. Please insert code below to ensure the desired behavior. Please remember to declare and initialize the conditional variable too.

```
static double rn=0.5;
static pthread_mutex_t rn_lock = PTHREAD_MUTEX_INITIALIZER;
(static) pthread_cond_t cond = PTHREAD_COND_INITIALIZER;

void *f(void *arg)
{
  pthread_mutex_lock(&rn_lock);

  while(rn <= 1.0)
     pthread_cond_wait(&cond, &rn_lock);

  rn -= 1.0;

  pthread_mutex_unlock(&rn_lock);

  return NULL;
}
```

2. (2 points) Which function is used to signal a conditional variable:
   (a) pthread_signal
   (b) signal
   (c) sigaction
   (d) pthread_cond_signal

   D.

3. (2 points) Assume that a thread is blocked on a call to *pthread_cond_wait*. Which of the following is true.
   (a) It releases the associated mutex lock when it is blocked, and re-acquires it when *pthread_cond_wait* returns.
   (b) It releases the associated mutex lock when it is blocked, and does *not* re-acquire it when *pthread_cond_wait* returns.
   (c) It holds the associated mutex lock for the entire duration that it is blocked, and continues to hold it after *pthread_cond_wait* returns.
   (d) It holds the associated mutex lock for the entire duration that it is blocked, but releases it when *pthread_cond_wait* returns.

   A.