**Name:** Sean Lossef **RCS ID:** losses @rpi.edu

**CSCI 4210 — Operating Systems** ⇝⇝⇝
**Spring 2020 Quiz 6 (April 6, 2020)**

- Please silence and put away all laptops, notes, books, phones, electronic devices, etc.
- This quiz is designed to take 25 minutes; therefore, for 50% extra time accommodations, the expected time is 38 minutes and 100% extra time accommodations is 50 minutes (i.e., the end of class).
- Questions will not be answered except when there is a glaring mistake or ambiguity in a question. Please do your best to interpret and answer each question.

**Honor Pledge:** On my honor, I have neither given nor received any unauthorized aid on this quiz.

1. **(6 POINTS)** Which output cannot occur in the given code? Note that there are no compilation warnings or errors in the given code. The `#include` directives are omitted to save space on the page. Assume that all system calls succeed. Circle the best answer.

```
void * foolish( void * arg )
{
  char * s = calloc( strlen( arg ) + 1, sizeof( char ) );
  strcpy( s, arg + 2 );
  fprintf( stderr, "%s", s );
  free( s );
  return NULL;
}

int main()
{
  pthread_t tid;
  char * q = "DON'T FORGET ABOUT THE EXAM!";
  pthread_create( &tid, NULL, foolish, q + 4 );
  fprintf( stderr, "%c%c TO%s", q[9], q[15], q + 18 );
  return EXIT_SUCCESS;
}
```

(a) `GO TO THE EXAM!`
(b) `GO TO THE EXAM!FORGET ABOUT THE EXAM!`
(c) `FORGET ABOUT THE EXAM!GO TO THE EXAM!`
(d) `FORGET ABOUT THE EXAM!`
(e) All outputs can occur.

2. **(3 POINTS)** When you call `pthread_detach()`, what happens? Circle the best answer.

   (a) The thread detaches by terminating and returning `NULL`
   (b) A child thread is acknowledged as being terminated
   (c) Memory is copied from one thread to another, i.e., they join forces
   (d) The thread disconnects from its parent thread and no longer needs to be joined
   (e) A new thread is created within the running process
   (f) The thread creates a shared memory segment

For the next two questions (3 and 4) consider the following code:

```
n = recvfrom( sd, buffer, MAXBUFFER, 0, (struct sockaddr *) &client,
              (socklen_t *) &len );
printf( "Rcvd datagram from %s port %d\n",
  <XXXX>( client.sin_addr ), ntohs( client.sin_port ) );
```

3. **(3 POINTS)** Which function should be used in place of `<XXXX>` above? Circle the best answer.

   (a) sendto()
   (b) inet_ntoa()
   (c) inet_aton()
   (d) socket()

4. **(3 POINTS)** The function ntohs() performs what kind of operation? Circle the best answer.

   (a) Internet communication
   (b) Error checking
   (c) Threading
   (d) Data marshalling
   (e) None of the above

5. **(5 POINTS)** The `P()` operation for semaphores is shown below. Identify the line or lines that comprise the critical section where a context switch can compromise the semaphore. Write the lines below.

```
1 P( semaphore S)
2 {
3   while (S==0)
4   {
5   }
6   S--;
7 }
```

Line(s): 4-5: A context switch that occurs between S==0 deemed false and s-- could change the integrity of the value of s