ICS 215 Final
closed book, closed notes, closed electronics (computer, cell phone, etc.)

1. Questions from Quiz Shows (5 pts)
   a) In Python, give an example of defining a new regular expression object that exactly matches the string "Python" only.
      ```python
      re.compile("^Python$")
      ```
   
   b) In Python, what is the main difference between a list and a tuple?
      
      You cannot change the components of a tuple (or You cannot use indices to access a tuple)
   
   c) What is the equivalent of Perl's foreach in Python?
      ```python
      for var in list:
      ```
   
   d) What is one main advantage of PHP over Perl for server-side scripting of dynamic web pages?
      PHP code can be interspersed freely with HTML
   
   e) In PHP, how can you tell what is a variable?
      It always begins with a $
   
   f) In JavaScript, how do you redirect the current window to a different location?
      ```javascript
      window.location="url.html"
      ```
   
   g) In JavaScript, how would you get the first 9 digit substring of the string str?
      ```javascript
      str.match(/\d{9}/)[0]
      ```
   
   h) In Perl, write a search program fragment to match periods followed by one or more white space characters followed by a capitalized word.
      ```perl
      /^\s+[A-Z]+$/
      ```
   
   i) In Perl, what is the simplest way to set the value of the key 'chin' to be 'David N. Chin' in the hash %uhunix_accounts?
      ```perl
      $uhunix_accounts{'chin'} = 'David N. Chin';
      ```
   
   j) Name two key differences between scripting languages and systems programming languages.
      Scripting languages are typeless (vs. typed), interpreted (vs. compiled), and higher level.
2. Better Web App Development (2 pts)
In Sean Kelly’s screencast comparing different web development languages and frameworks:

a) What was the problem with developing user interfaces in C++ (be specific)?

C++ did not have fast turnaround time because it took too long to modify the code, compile the code and then run it.

b) What did he call having to set lots of parameters in XML files?

XML situps

3. Perl scripting (3 pts)
Write a Perl subroutine called unique_name that takes as arguments a filename and a directory path that ends in the correct directory separator (“/” or “\”) and returns a non-conflicting filename. If the filename does not exist in the directory path, then the original filename is returned. If it does exist, modify the filename by adding different integers (starting from 1) to the base of the filename until a filename is found that does not exist in the path, e.g. document.doc would become document1.doc, then document2.doc, etc.

```perl
sub unique_name {
  my ($filename, $path) = @_;  
  return $filename if (!–e "$path$filename");  
  $filename =~ /^(.*)\.(\[^.]*$)/;  
  my ($filename_base, $filename_extension) = ($1, $2);  
  $i = 1;  
  while(1) {
    $new_name = "$filename_base$i.$filename_extension";
    return $new_name if (!–e $new_name);
    $i++;
  }
}
```
JavaScript (3 pts)
Write an HTML page with embedded JavaScript that has a form with a text type input box for entering a social security number, a textarea for comments and a submit button. When the user changes the social security number, the program checks that it is in the form ddd-dd-dddd with no extra stuff before or after and pop up an alert if incorrect.

```html
<html>
<head>
<script language="JavaScript" type="text/javascript">
function validate(ssn) {
   if (! ssn.match(/^\d{3}-\d{2}-\d{4}$/)) {
      window.alert('Invalid Social Security Number. Must be 9 digits in the form ddd-dd-dddd.');
      return false;
   }
   return true;
}
</script>
</head>
<body>
<form name="form">
   Social security number: <input type="text" name="ssn" onChange="validate(this.value);">
   <br />
   Comments: <textarea name="comments"></textarea>
   <br />
   <input type="submit">
</form>
</body>
</html>
```
4. **PHP Scripting (6 pts)**

Write a PHP program, register.php, that initially has a text type input box for the user’s name, a radio button for three different dates 6/1, 6/2, or 6/3 and a Submit button with an action that is the same register.php. If the user has submitted successfully (name is not empty and one date is selected), then the program says only “Thank you *name* for registering for the seminar on *date*.”

```html
<html>
<head></head>
<body>
<?php
if(!empty($_REQUEST['name']) && isset($_REQUEST['date'])) {

<p>Thank you <?php echo $_REQUEST['name']; ?> for registering for the seminar on <?php print $_REQUEST['date']; ?></p>
<?php
} else {

<form name="form" action="register.php" method="get">
Name: <input type="text" name="name">
<br />
<input type="radio" name="date" value="6/1" /> 6/1
<input type="radio" name="date" value="6/2" /> 6/2
<input type="radio" name="date" value="6/3" /> 6/3
<br />
<input type="submit" />
</form>

<?php
}
?>
</body>
</html>
```
5. **Python Scripting (6 pts)**

Write a Python class Rectangle whose __init__ takes two numbers (the length and width), has a @property called area that returns the product of the length and width, and overloads the __add__ operator so that if the other argument is also a Rectangle, then a new rectangle is returned with length equal to the sum of the lengths of the added Rectangles and likewise for the width. Finally, write a generator called rectangle_multiples that takes a Rectangle as an argument and yields new Rectangles that are positive integer multiples of the original Rectangle with length progressing from \( l \) to \( 2l \), to \( 3l \), to \( 4l \) etc. and likewise for the width. The first call to next() should return a Rectangle identical in length and width to the argument.

```python
class Rectangle(object):
    def __init__(self, len, wid):
        self.length = len
        self.width = wid

    @property
    def area(self):
        return self.length * self.width

    def __add__(self, other):
        if isinstance(other, Rectangle):
            return Rectangle(self.length + other.length, self.width + other.width)

def rectangle_multiples(rect):
    length = rect.length
    width = rect.width
    n = 1
    while True:
        yield Rectangle(n * length, n * width)
        n += 1
```