1. (14 pts total) History of computation

(a) (4 pts) Match each item in the left column with the most relevant item in the right column.

<table>
<thead>
<tr>
<th>Item</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jacquard’s loom</td>
<td>6</td>
</tr>
<tr>
<td>ENIAC</td>
<td>4</td>
</tr>
<tr>
<td>Moore’s Law</td>
<td>7</td>
</tr>
<tr>
<td>Difference Engine</td>
<td>3</td>
</tr>
<tr>
<td>Harvard Mark I</td>
<td>1</td>
</tr>
<tr>
<td>Grace Hopper</td>
<td>8</td>
</tr>
<tr>
<td>Ada Lovelace</td>
<td>5</td>
</tr>
<tr>
<td>Alan Turing</td>
<td>2</td>
</tr>
<tr>
<td>1. Electromechanical computer</td>
<td></td>
</tr>
<tr>
<td>2. Enigma cipher</td>
<td></td>
</tr>
<tr>
<td>3. Polynomial function</td>
<td></td>
</tr>
<tr>
<td>4. Vacuum tubes</td>
<td></td>
</tr>
<tr>
<td>5. First programmer</td>
<td></td>
</tr>
<tr>
<td>6. Hollerith tabulating machine</td>
<td></td>
</tr>
<tr>
<td>7. Exponential function</td>
<td></td>
</tr>
<tr>
<td>8. Debugging</td>
<td></td>
</tr>
</tbody>
</table>

(b) (6 pts) We want to use the method of finite differences to create a table of numbers for the function \( f(x) = 5x^2 - 4x + 3 \). Compute the necessary difference functions and the initial values for \( x = 0 \) for this machine. Then fill in the table for \( x \) values from 1 to 3.

\[
\Delta f(x) = \quad 10 \quad x + \quad 1
\]

\[
\Delta^2 f(x) = \quad 10
\]
2. This problem focuses on expressions and data types.

(a) (6 pts) For each of the following Ruby expression, write down the value that would be output if the expression was evaluated in irb.

40 / 9     _____4_______  15.0 /2  _____7.5_______
2 * 2 ** 4  _____32_______  15 % 2  _______1_____
6 + 4 * 2 - 1  ______13______  2 != 2  _______false_____

(b) (2 pts) Write a Ruby function triangle_area that takes two parameters h and b, respectively, for the height and base of a triangle, and returns the area of the triangle given by the formula

\[
A = \frac{1}{2} \text{ height } \times \text{ base}
\]

```ruby
def triangle_area(h, b)
    return 0.5 * h * b
end
```
(c) (2 pts) Write a Ruby function `truncated_triangle` that takes height and base parameters (h and b) as input, and computes the area of a triangle that the tip cut off. The tip is also a triangle; its height and base are 10% of the height and base, respectively, of the larger triangle, as shown in the figure. Use the `triangle_area` function in your solution.

```ruby
def truncated_triangle area (h, b)
    h_tip = 0.1 * h
    b_tip = 0.1  * b
    return   triangle_area(h, b) – triangle_area(h_tip, b_tip)
end
```

(d) (2 pts)

```ruby
def mystery1(m, n)
i = 0
    while i <= n-1 do
        i = i + 1
        print i ** m, " "
    end
end
```

The Ruby function above prints a sequence of numbers. Which of the following is the output of the function expressed in terms of m and n? Circle your answer.

1     2m    3m  …  nm            OR                   1    m2     m3    …    mn

The first one
(e) (4 pts) If the print statement was taken outside of the while loop to occur right after the while statement, as shown below, what would the function call mystery2(2, 3) print?

```ruby
def mystery2(m, n)
  i = 0
  while i <= (n-1) do
    i = i + 1
  end
  print i ** m, " "
end
```

_____9______________________________

(f) (4 pts) Consider the following Ruby function:

```ruby
def mystery3(m, n)
  i = 0
  result = 0
  while i <= (n-1) do
    i = i + 1
    result = result + i ** m
  end
  return result
end
```

What would the value of the variable x be after executing the following assignment statement below?

```ruby
x = mystery3(2,4)
```

_____2 + 4 +8+16 = 30______________________________

(3) (20 pts total) This question focuses on the array data type and iterators.

(a) (6 pts) Assume the following list definition in Ruby using an array.

```ruby
cars = ["Honda", "Toyota", "Kia", "Chrysler", "Mercedes"]
```

What would be displayed in irb for each of the following Ruby expressions?

```ruby
cars.length
```

_________5____________

```ruby
cars.first
```

_________"Honda"__________
cars[1]  
    "Toyota"

cars.include?("Mazda")  
    false

cars.include?("KIA")  
    false

(b) (4 pts) Assume the following list definition in Ruby using an array.

```ruby
a = [1, 2, [3, 4, 5], 6]
```

What would be displayed in irb for each of the following Ruby expressions?

- `a.length`  
  4
- `a.first`  
  1
- `a[2]`  
  `[3, 4, 5]`
- `a + a`  
  `[1, 2, [3, 4, 5], 6, 1, 2, [3, 4, 5], 6]`

(c) (10 pts) Assume the following list definition in Ruby using an array.

```ruby
a = [2, 4, 6, 7, 8]
```

What would be returned in irb for if the following Ruby expressions are executed in the given order?

- `a.collect{ |x| x / 2 }`  
  `[1, 2, 3, 3, 4]`
- `a`  
  `[2, 4, 6, 7, 8]`
- `a.select{|x| x.even?}`  
  `[2, 4, 6, 8]`
- `a.delete_if{ |x| x % 3 == 0 }`  
  `[2, 4, 7, 8]`

**Hint:** `delete_if` is a **destructive** method in Ruby.

```
```
4. (20 pts total) This question focuses on looping.

(a) (8 pts) We wish to define a Ruby function `out_of_order` that takes an “almost sorted” list as input and returns the first item that is not in ascending order. The function should return nil if the list is entirely in ascending order. For example, `out_of_order([1, 5, 17, 12, 24])` should return 12, since 12 is less than the preceding item, 17. Complete the following iterative function `out_of_order`.

```ruby
def out_of_order(list)
  index = __0_____
  while index < _____list.length-1___
    if ___list[index]_____________ > _____list[index+1]_____________
      return _____list[index+1]_______
    end
    index = ___index + 1_________
  end
  return ___nil______
end
```

(b) (8 pts) Consider the following recursive algorithm for returning the first item in a list that is not in ascending order, else nil. Complete the recursive definition of `out_of_order`.

1. If the list has fewer than two elements, return nil.
2. If the first element in the list is greater than the second element, return the first element.
3. Otherwise return the result of a recursive call on the tail of the list (i.e., everything beyond the first element.)
def out_of_order(list)
    if list.length < 2 then
        return nil
    elsif list[0] > list[1] then
        return list[0]
    else
        return out_of_order(list[1..list.length-1])
    end

(c) (2 pts) Give an example of a six element list that would be a worst case input for out_of_order.

    _____Any six element list with all elements in nondescending order________________________

(d) (2 pts) What is the big O worst case complexity of out_of_order?

    ______O(n)_______________________________

5. (20 pts) This question deals with searching and sorting.

(a) (2 pts)

What is the big O complexity of binary search? ______O(log n)________

What is the big O complexity of insertion sort? ______O(n^2)________

(b) (6 pts) Fill in the table below to show how binary search would locate the value “e” in the list ["a", "b", "c", "d", "e", "f", "g", "h", "i", "j", "k"]. Use the binary search algorithm taught in the book and covered in lecture. Note: this table may contain extra rows.

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Low</th>
<th>High</th>
<th>Mid</th>
<th>list[mid]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
(c) (6 pts) For each sorting algorithm described below, give its correct name:

- For each input item, find its proper position in the result list and add it at that position.
  
  ______insertion sort____________________

- For each position \( i \) in the list, find the index of the smallest item at or to the right of position \( i \), and swap \( \text{list}[i] \) with \( \text{list}[\text{index of smallest}] \).

  ______selection sort____________________

- Organize the inputs into \( N \) groups of size 1. Systematically combine adjacent groups to form \( N/2 \) sorted groups, each of size 2. Repeat the process, combining adjacent groups of size 2 to form \( N/4 \) sorted groups of size 4. Keep going until you have one sorted group of size \( N \).

  ________merge sort______________________

(d) (6 pts) Suppose we want to know if all the elements of a list are the same. For example, 

\[
\text{all_same}([1, 1, 1, 99, 1, 1]) \text{ should return false, but }
\text{all_same}(["f", "f", "f", "f"])) \text{ should return true. Here are two solutions. Fill in the missing elements.}
\]

```python
def all_same1(list):
    sorted_list = list.sort
    if sorted_list[0] == sorted_list[ __list.length -1__________ ] then
```

```python
def all_same2(lst):
    return lst[0] == lst[-1]
```
return ____true____
else
  return ____false____
end
end

def all_same2(list)
  list.each { |x| return ____false____ if ___list[x]___ != list[0] }
  return ____true____
end

What is the big O complexity of all_same1?  ______0(1)_if we could assume that sort is a constant time operation but it is not. The complexity is hidden in sort. The best runtime for any sort is O(n log n), O(n^2) would also be acceptable.

What is the big O complexity of all_same2?  ______0(n)__________

6. (6 pts) This question is based on your readings from the book *Blown to Bits*.

When you print a report using a laser printer, can you assume that no one can tell who printed it? Give a yes/no answer followed by a one sentence explanation.

No. Many color printers secretly encode the printer serial number, date, and time on every page they print.