Walter Blair

Part 1

```
def sum(arr)
  sum = 0
  arr.each do |x| sum += x; end;
  sum
end

def max_2_sum arr
  arr.sort!.reverse!
  if arr[1]
    return arr[0] + arr[1]
  elsif arr[0]
    return arr[0]
  else return 0
  end
end

def sum_to_n? arr, n
  if arr.length < 2
    return false
  else
    index = 0
    arr.each do |i| index+=1
      arr.drop(index).each do |j|
        return true if i+j==n
      end
    end
    return false
  end
end
```

Part 2
```
# References - rubular.com
def hello(name)
  return "Hello, #{name}"
end

def starts_with_consonant? s
  return !s.nil? && (s[0] =~ /[a-zA-Z]/) && !'aeiouAEIOU'.include?(s[0])
end
```

```ruby
def binary_multiple_of_4? s
  if s.length == 1
    return s == "0"
  else
    return s !~ /\D/ && s[-2..-1] == "00"
  end
end
```

Part 3

```ruby
# References - https://stackoverflow.com/questions/10459901/how-do-you-round-a-float-to-two-
decimal-places-in-jruby
class BookInStock
  attr_accessor :isbn
  attr_accessor :price

  def initialize(isbn, price)
    throw ArgumentError if isbn.empty? || price <= 0
    @isbn = isbn
    @price = price
  end

  def price_as_string()
    "$" + sprintf('%.2f', price)
  end
end
```

```
walter@walter-Gazelle:~/Desktop/Fall 2017/602/hw-ruby-intro$ rspec

Ruby intro part 1
  #sum
    should be defined
    returns correct sum [20 points]
    works on the empty array [10 points]
  #max_2_sum
    should be defined
    returns the correct sum [7 points]
    works even if 2 largest values are the same [3 points]
    returns zero if array is empty [10 points]
    returns value of the element if just one element [10 points]
  #sum_to_n
    should be defined
    returns true when any two elements sum to the second argument [30 points]
    returns false for any single element array [5 points]
    returns false for an empty array [5 points]

#hello
  should be defined
  The hello method returns the correct string [30 points]

#starts_with_consonant?
  should be defined
  classifies true cases [10 points]
  classifies false cases [10 points]
  works on the empty string [5 points]
  works on nonletters [5 points]

#binary_multiple_of_4?
  should be defined
  classifies valid binary numbers [30 points]
  rejects invalid binary numbers [10 points]

BookInStock
  should be defined
  getters and setters
    should set ISBN [10 points]
    should set price [10 points]
    should be able to change ISBN [10 points]
    should be able to change price [10 points]
  constructor
    should reject invalid ISBN number [10 points]
    should reject zero price [10 points]
    should reject negative price [10 points]
  #price_as_string
    should be defined
    should display 33.95 as "$33.95" [10 points]
    should display 1.1 as $1.10 [10 points]
    should display 20 as $20.00 [10 points]

Finished in 0.00881 seconds (files took 0.07815 seconds to load)
34 examples, 0 failures

walter@walter-Gazelle:~/Desktop/Fall 2017/602/hw-ruby-intro$
```