

3. (5 pts) Write a Ruby method *find_course_num* that given a string *str*, uses regular expressions and back references to find the first course number in the string. A course number has 4 uppercase letters, followed by a 3 digit number. The method should return the course number as an integer value, or *nil* if no course number is found.

Examples: `find_course_num("CMSC131 CMSC330")` # returns 131
`find_course_num("CMsC131 CMSC330")` # returns 330
`find_course_num("CMsC131 CMS330")` # returns *nil*

```
def find_course_num(str)
  if str =~ /[A-Z]{4}([0-9]{3})/
    return $1.to_i
  else
    return nil
  end
end
```

Some helpful functions (not all need to be used)	
<code>a.each { ... }</code>	// apply code block to each element in array
<code>puts b</code>	// print b followed by a newline
<code>s.to_i</code>	// returns integer value for string s
<code>n.to s</code>	// returns string for integer n

```
def find_course_num(str)
  if str =~ /[A-Z][A-Z][A-Z][A-Z](\d\d\d)/
    return $1.to_i
  else
    return nil
  end
end
```

4. (5 pts) Write a Ruby method *print_str_array* that given an array of strings, uses the `Array.each` method and a code block to print out each string on a separate line, prefixed by the index of the string in the array, separated by a single space.

Examples: `print_str_array(["terrapin","tortoise","turtle"])` # prints
 0 terrapin
 1 tortoise
 2 turtle

```
def print_str_array(arr)
  idx = 0
  arr.each { |x| puts "#{idx} #{x}" ; idx += 1 }
end
```

```
def print_str_array(arr)
  idx = 0
  arr.each { |x| puts idx.to_s + " " + x ; idx += 1 }
end
```

partial credit

```
def print_str_array(arr) idx = (0..(arr.size-1)).each { |x| puts "#{x} #{arr[x]}" } end
def print_str_array(arr) arr.size.times { |x| puts "#{x} #{arr[x]}" } end
```