Ex:

```
function c = pattern(vec, binum)

  c = [ ];
  len = length(binum);

  for index = 1:length(vec) - len + 1
    if vec(index:index + len - 1) == binum
      c = [c, index];
    end
  end
end
```

For the above Matlab® function, write down exactly what Matlab® prints out in response to the following commands:

```
>> inbin = [0, 1, 1, 0, 0, 1, 1, 0];
>> patt = [1, 1, 0];
>> pattern(inbin, patt)
```

Sol'n: This function looks for the pattern `patt = [1, 1, 0]` in `inbin = [0, 1, 1, 0, 0, 1, 1, 0]`. The function steps thru `inbin` from left to right looking for [1, 1, 0]. When the pattern is found, the index in `inbin` where the pattern starts is saved in array `c`.

By examining `inbin`, we see that [1, 1, 0] will be found at positions 2 and 6. These values are displayed when the function returns. Since no variable was set equal to `pattern(inbin, patt)`, the resulting value is printed as "`ans`":

```
ans =
     2     6
```