Problem 1. (20 pts) What does the following program output when run?

```cpp
#include <iostream>

int main()
{
    int *p;
    int q = 20;
    p = &q;
    int *r;
    cout << "p: " << *p << endl; //Answer p:_______________________
    r=p;
    cout << "r: " << *r << endl; //Answer r:_______________________
    *p = 2 * *p;
    cout << "q: " << q << endl; //Answer q:_______________________
    *p = 2 * *p;
    cout << "s: " << s << endl; //Answer s:_______________________
}
```
Problem 2. (25 pts) Write a function **bitmap** that accepts a vector \( x \) of integers and an integer value \( y \), and returns a new vector which replaces all values in \( x \) with either 0 or 1 depending on whether \( x[i] < y \) or \( x[i] \geq y \), respectively. (i.e. The returned vector contains 1 in position \( i \) if \( x[i] \geq y \) and contains a 0 in position \( i \) otherwise - if \( x[i] < y \).)

Imagine your function in the following context:

```cpp
#include <iostream>
#include <vector>

vector<int> bitmap(vector<int> x, int y);
{
    YOUR PROGRAM HERE!
}

int main()
{
    vector<int> nums;
    nums.pushback(9);
    nums.pushback(27);
    nums.pushback(14);
    nums.pushback(35);
    nums.pushback(16);

    vector<int> new_nums;

    new_nums = bitmap(nums, 25); // Calling bitmap
}
```

After calling `bitmap` in the above program, the value of the vector `new_nums` should be

```cpp
new_nums[0] = 0;
new_nums[1] = 1;
new_nums[2] = 0;
new_nums[3] = 1;
new_nums[4] = 0;
```
Problem 3. (25 pts total) A two-dimensional point consists of an \( x \)-coordinate value and a \( y \)-coordinate value. Give the member functions for the Point class below:

```cpp
class Point
{
public:
  Point(); // creates the point (0,0)
  Point(int x, int y); // creates the point (x,y)
  int getX(); // returns the x coordinate
  int getY(); // returns the y coordinate
  void move(int dx, int dy); // moves the point (x,y) to (x+dx,y+dy)

private:
  int xcoordinate;
  int ycoordinate;
};
```
Problem 4. (30 pts total) A polygon is a closed plane figure having three or more straight sides. A polygon can be defined by the vertices (corner points). Class Polygon is defined as follows:

```cpp
class Polygon
{
    public:
        Polygon(); // empty polygon - no points
        Polygon(vector<Point> p); // polygon defined by vector p
        void move(int dx, int dy); // moves all the corners dx on x-axis and dy on y axis

    private:
        vector<Point> corners;
};
```

Using the class Point from Problem 3, provide solutions to the following:

(a) Implement the constructor `Polygon(vector<Point> p)`.

(b) Implement the member function `move` which moves each corner of the polygon dx units on the x axis and dy units on the y axis.

(c) Write an `int main()` function that creates a quadrilateral polygon with points at (0,0), (0,1), (1,1) and (1,0) and then moves the quadrilateral 3 units in the x-direction and 5 units in the y-direction.