Programming Design, Spring 2016
Homework 8 Solution

Solution provider: Wei-Hung Liao
Department of Information Management
National Taiwan University

Problem 1

(a) Yes, they are the same. The only different part between the two functions is `arr[i][j]` and `*(*(arr+i)+j)`. Since `arr[i]` is the more convenient way to represent `*(arr+i)` and so do `arr[i][j]` and `*(*(arr+i)+j)`, we can recognize that the two functions are the same.

(b) `*p` is equal to `*(arr+i)` and `arr[i]` without any modification in each iteration in the `for` loop, the two functions are obviously different.

(c) The modified program is below:

```cpp
int main()
{
  int r = 3;
  int** array = new int*[r];
  for(int i = 0; i < r; i ++)
  {
    array[i] = new int[i + 1];
    for(int j = 0; j <= i; j ++)
      array[i][j] = j + 1;
  }
  print(array ,r);
  for(int i = 0; i < r; i++)
  {
    delete[] array[i];
    array[i] = nullptr;
  }
  delete[] array;
  array = nullptr;
  return 0;
}
```

(d) If you want to pass a static two-dimensional array to the function, you should always tell the function the size of each one-dimensional array, or the function won't know how to change the address when the first dimension is changed. So the compiler treats `int** array` and `int array[3][3]` in different way.

Problem 2

Please see the attached CPP file.

Problem 3

Please see the attached CPP file.