Pointer Examples #2

More examples with pointers:

Example 1.6

```
//
// New and Delete
//
#include <iostream>
using namespace std;

int main (){

    int *p;

    p = new int;
    *p = 11;
    cout << *p << endl;
    delete p;

    p = new int;
    *p = 13;
    cout << *p << endl;
    delete p;

    p = new int;
    *p = 17;
    cout << *p << endl;
    delete p;
}
```

Example 1.7

```
//
// Dynamically allocated arrays
//
```

```
#include <iostream>
#include <string>
using namespace std;

class Employee{
public:
    string name;
    int sal;
};

int main (){

    Employee *emp;

    emp = new Employee [ 3 ];

    emp[0].name = "Stan";
    emp[1].name = "Kyle";
    emp[2].name = "Kenny";

    cout << emp[0].name << endl;
    cout << emp[1].name << endl;
    cout << emp[2].name << endl;

    delete [] emp;
}
```

Example 1.8

```
//
// Arrays and Pointers are twins.
//
#include <iostream>
using namespace std;

int main (){
    int nums[5];
    int *m;
```

```
    cout << &nums[0] << endl;
    cout << nums << endl;

    m = nums;

    for (int k=0; k<5; k++){
         m[k] = k*5;
    }

    for (int k=0; k<5; k++){
         cout << nums[k] << endl;
    }
}
```

Example 1.9

```
//
// 2Dimensional Dynamic arrays
//
#include <iostream>
using namespace std;

int main (){

    int **matrix;

                                  // Alocating a 5 by 7 2D Array
    matrix = new int* [5];
    for (int k=0; k<5; k++){
        matrix[k] = new int[7];
    }

                                  // De-allocating
    for (int k=0; k<5; k++){
        delete [ ] matrix[k];
    }
    delete [ ] matrix;
```

```
}
```