RETURN ARRAY FROM FUNCTIONS IN C++

C++ does not allow to return an entire array as an argument to a function. However, you can return a pointer to an array by specifying the array's name without an index.

If you want to return a single-dimension array from a function, you would have to declare a function returning a pointer as in the following example:

```
int * myFunction()
{
.
.
.
}
```

Second point to remember is that C++ does not advocate to return the address of a local variable to outside of the function so you would have to define the local variable as **static** variable.

Now, consider the following function, which will generate 10 random numbers and return them using an array and call this function as follows:

```
#include <iostream>
#include <ctime>

using namespace std;

// function to generate and retrun random numbers.
int * getRandom( )
{
  static int  r[10];

  // set the seed
  srand( (unsigned)time( NULL ) );
  for (int i = 0; i < 10; ++i)
  {
    r[i] = rand();
    cout << r[i] << endl;
  }

  return r;
}

// main function to call above defined function.
int main ()
{
   // a pointer to an int.
   int *p;

   p = getRandom();
   for ( int i = 0; i < 10; i++ )
   {
       cout << "*(p + " << i << ") : ";
       cout << *(p + i) << endl;
   }

   return 0;
}
```

When the above code is compiled together and executed, it produces result something as follows:

```
624723190
1468735695
807113585
```

```
976495677
613357504
1377296355
1530315259
1778906708
1820354158
667126415
*(p + 0) : 624723190
*(p + 1) : 1468735695
*(p + 2) : 807113585
*(p + 3) : 976495677
*(p + 4) : 613357504
*(p + 5) : 1377296355
*(p + 6) : 1530315259
*(p + 7) : 1778906708
*(p + 8) : 1820354158
*(p + 9) : 667126415
```