```
...inters\02-Arrays\01-ArraysAndPointers\ArraysAndPointers.c
```

```
1
```

```
1 #include <stdio.h>
 2
 3 int main(void)
 4
 5
         //variable declarations
         int iArray[] = { 12, 24, 36, 48, 60, 72, 84, 96, 108, 120 };
 6
 7
         float fArray[] = { 9.8f, 8.7f, 7.6f, 6.5f, 5.4f };
 8
         double dArray[] = { 1.222222, 2.333333, 3.444444 };
         char cArray[] = { 'A', 'S', 'T', 'R', 'O', 'M', 'E', 'D', 'I', 'C', 'O', 'M', >
 9
           'P', '\0' };
10
11
         //code
12
         printf("\n\n");
13
         printf("Integer Array Elements And The Addresses They Occupy Are As Follows : 🤝
14
         printf("iArray[0] = %d
                                    \t At Address : %p\n", *(iArray + 0), (iArray + 0));
         printf("iArray[1] = %d
                                    \t At Address : %p\n", *(iArray + 1), (iArray + 1));
15
                                    \t At Address : %p\n", *(iArray + 2), (iArray + 2));
         printf("iArray[2] = %d
16
                                    \t At Address : %p\n", *(iArray + 3), (iArray + 3));
\t At Address : %p\n", *(iArray + 4), (iArray + 4));
17
         printf("iArray[3] = %d
18
         printf("iArray[4] = %d
                                    \t At Address : %p\n", *(iArray + 5), (iArray + 5));
19
         printf("iArray[5] = %d
                                    \t At Address : %p\n", *(iArray + 6), (iArray + 6));
         printf("iArray[6] = %d
20
                                    \t At Address : %p\n", *(iArray + 7), (iArray + 7));
         printf("iArray[7] = %d
21
         printf("iArray[8] = %d
                                    \t At Address : %p\n", *(iArray + 8), (iArray + 8));
22
23
         printf("iArray[9] = %d
                                    \t At Address : %p\n", *(iArray + 9), (iArray + 9));
24
25
         printf("\n\n");
26
         printf("Float Array Elements And The Addresses They Occupy Are As Follows : \n >
           \n");
27
         printf("fArray[0] = %f
                                    \t At Address : %p\n", *(fArray + 0), (fArray + 0));
         printf("fArray[1] = %f
                                    \t At Address : %p\n", *(fArray + 1), (fArray + 1));
28
                                    \t At Address : %p\n", *(fArray + 2), (fArray + 2));
         printf("fArray[2] = %f
29
                                    \t At Address : %p\n", *(fArray + 3), (fArray + 3));
         printf("fArray[3] = %f
30
         printf("fArray[4] = %f \t At Address : %p\n", *(fArray + 4), (fArray + 4));
31
32
         printf("\n\n");
33
34
         printf("Double Array Elements And The Addresses They Occupy Are As Follows :
           n'n;
         printf("dArray[0] = %lf \t At Address : %p\n", *(dArray + 0), (dArray + 0));
35
         printf("dArray[1] = %lf \t At Address : %p\n", *(dArray + 1), (dArray + 1));
36
         printf("dArray[2] = %lf \t At Address : %p\n", *(dArray + 2), (dArray + 2));
37
38
39
         printf("\n\n");
40
         printf("Character Array Elements And The Addresses They Occupy Are As
           Follows : \n\n");
41
         printf("cArray[0] = %c
                                    \t At Address : %p\n", *(cArray + 0), (cArray + 0));
                                     \t At Address : %p\n", *(cArray + 1), (cArray + 1));
         printf("cArray[1] = %c
42
                                    \tag{carray + 1), (carray + 1), \tag{carray + 1), \tag{carray + 1), \tag{carray + 2}, (carray + 2)); \tag{carray + 2}, (carray + 2)); \tag{carray + 3}, (carray + 3)); \tag{carray + 4}, (carray + 4); \tag{carray + 5}, (carray + 5); \tag{carray + 5}, (carray + 5); \tag{carray + 5}, \tag{carray + 5});
         printf("cArray[2] = %c
43
         printf("cArray[3] = %c
44
45
         printf("cArray[4] = %c
46
         printf("cArray[5] = %c
         printf("cArray[6] = %c \ \ t \ At \ Address : %p\n", *(cArray + 6), (cArray + 6));
47
```

```
...inters\02-Arrays\01-ArraysAndPointers\ArraysAndPointers.c
      48
49
      printf("cArray[9] = %c \t At Address : %p\n", *(cArray + 9), (cArray + 9));
50
      printf("cArray[10] = %c \t At Address : %p\n", *(cArray + 10), (cArray +
51
        10));
      printf("cArray[11] = %c \t At Address : %p\n", *(cArray + 11), (cArray +
52
        11));
      printf("cArray[12] = %c \t At Address : %p\n", *(cArray + 12), (cArray +
53
      printf("cArray[13] = %c \t At Address : %p\n", *(cArray + 13), (cArray +
54
        13));
55
      return(0);
56
57 }
58
59
```

60