```
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 };
       double dArray[] = { 1.222222, 2.333333, 3.444444 };
 8
 9
       char cArray[] = { 'A', 'S', 'T', 'R', 'O', 'M', 'E', 'D', 'I', 'C', 'O', 'M', →
         'P', '\0' };
10
       //code
11
12
       printf("\n\n");
13
       printf("Integer Array Elements And The Addresses They Occupy Are As Follows : >
         \n\n");
       printf("iArray[0] = %d \t At Address : %p\n", *(iArray + 0), (iArray + 0));
14
       printf("iArray[1] = %d \t At Address : %p\n", *(iArray + 1), (iArray + 1));
15
       printf("iArray[2] = %d \t At Address : %p\n", *(iArray + 2), (iArray + 2));
16
       printf("iArray[3] = %d \t At Address : %p\n", *(iArray + 3), (iArray + 3));
17
       printf("iArray[4] = %d \t At Address : %p\n", *(iArray + 4), (iArray + 4));
18
19
       printf("iArray[5] = %d \t At Address : %p\n", *(iArray + 5), (iArray + 5));
       printf("iArray[6] = %d \t At Address : %p\n", *(iArray + 6), (iArray + 6));
20
       21
22
       printf("iArray[9] = %d \setminus t Address : %p\n", *(iArray + 9), (iArray + 9));
23
24
25
       printf("\n\n");
       printf("Float Array Elements And The Addresses They Occupy Are As Follows : \n >
26
         \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
       printf("fArray[2] = %f \t At Address : %p\n", *(fArray + 2), (fArray + 2));
29
       printf("fArray[3] = %f \ \ t \ At \ Address : %p\n", *(fArray + 3), (fArray + 3));
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";
35
       printf("dArray[0] = %lf \t At Address : %p\n", *(dArray + 0), (dArray + 0));
       printf("dArray[1] = %lf \ \ 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));
42
       printf("cArray[1] = %c
                               \t At Address : %p\n", *(cArray + 2), (cArray + 2));
43
       printf("cArray[2] = %c
                                \t At Address : %p\n", *(cArray + 3), (cArray + 3));
44
       printf("cArray[3]
                         = %c
45
       printf("cArray[4] = %c
                                \t At Address : %p\n", *(cArray + 4), (cArray + 4));
       printf("cArray[5] = %c
                               \t At Address: p\n'', *(cArray + 5), (cArray + 5));
46
       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));
52
      printf("cArray[11] = %c \t At Address : %p\n", *(cArray + 11), (cArray +
                                                                            P
        11));
      printf("cArray[12] = %c \t At Address : %p\n", *(cArray + 12), (cArray +
53
        12));
       printf("cArray[13] = %c \t At Address : %p\n", *(cArray + 13), (cArray +
54
        13));
55
      return(0);
56
57 }
58
59
```

60