```c
#include <stdio.h>

int main(void)
{
    //variable declarations
    int iArray[] = { 12, 24, 36, 48, 60, 72, 84, 96, 108, 120 };
    float fArray[] = { 9.8f, 8.7f, 7.6f, 6.5f, 5.4f };
    double dArray[] = { 1.222222, 2.333333, 3.444444 };
    char cArray[] = { 'A', 'S', 'T', 'R', 'O', 'M', 'E', 'D', 'I', 'C', 'O', 'M', 'P', '\0' };

    //code
    printf("\n\n");
    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));
    printf("iArray[1] = %d  \t At Address : %p\n", *(iArray + 1), (iArray + 1));
    printf("iArray[2] = %d  \t At Address : %p\n", *(iArray + 2), (iArray + 2));
    printf("iArray[3] = %d  \t At Address : %p\n", *(iArray + 3), (iArray + 3));
    printf("iArray[4] = %d  \t At Address : %p\n", *(iArray + 4), (iArray + 4));
    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));
    printf("iArray[7] = %d  \t At Address : %p\n", *(iArray + 7), (iArray + 7));
    printf("iArray[8] = %d  \t At Address : %p\n", *(iArray + 8), (iArray + 8));
    printf("iArray[9] = %d  \t At Address : %p\n", *(iArray + 9), (iArray + 9));

    printf("\n\n");
    printf("Float Array Elements And The Addresses They Occupy Are As Follows : \n\n");
    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));
    printf("fArray[2] = %f  \t At Address : %p\n", *(fArray + 2), (fArray + 2));
    printf("fArray[3] = %f  \t At Address : %p\n", *(fArray + 3), (fArray + 3));
    printf("fArray[4] = %f  \t At Address : %p\n", *(fArray + 4), (fArray + 4));

    printf("\n\n");
    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));
    printf("dArray[1] = %lf  \t At Address : %p\n", *(dArray + 1), (dArray + 1));
    printf("dArray[2] = %lf  \t At Address : %p\n", *(dArray + 2), (dArray + 2));

    printf("\n\n");
    printf("Character Array Elements And The Addresses They Occupy Are As Follows : \n\n");
    printf("cArray[0]  = %c  \t At Address : %p\n", *(cArray + 0), (cArray + 0));
    printf("cArray[1]  = %c  \t At Address : %p\n", *(cArray + 1), (cArray + 1));
    printf("cArray[2]  = %c  \t At Address : %p\n", *(cArray + 2), (cArray + 2));
    printf("cArray[3]  = %c  \t At Address : %p\n", *(cArray + 3), (cArray + 3));
    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));
    printf("cArray[6]  = %c  \t At Address : %p\n", *(cArray + 6), (cArray + 6));
```

```
48      printf("cArray[7]  = %c  \t At Address : %p\n", *(cArray + 7), (cArray + 7));
49      printf("cArray[8]  = %c  \t At Address : %p\n", *(cArray + 8), (cArray + 8));
50      printf("cArray[9]  = %c  \t At Address : %p\n", *(cArray + 9), (cArray + 9));
51      printf("cArray[10] = %c  \t At Address : %p\n", *(cArray + 10), (cArray +
          10));
52      printf("cArray[11] = %c  \t At Address : %p\n", *(cArray + 11), (cArray +
          11));
53      printf("cArray[12] = %c  \t At Address : %p\n", *(cArray + 12), (cArray +
          12));
54      printf("cArray[13] = %c  \t At Address : %p\n", *(cArray + 13), (cArray +
          13));
55
56      return(0);
57  }
58
59
60
```