//

#include <stdio.h>

void pointerArray(int \*Fun\_startPtr, int \*Fun\_endPtr); // starting address and ending address of array

void main(void)

{

int intArray[10] = { 0 };

int \*startPtr = &intArray[0]; // starting address of array

int \*endPtr = &intArray[10]; // ending address of array

size\_t k;

for (k = 0; k < 10; k++)

{

intArray[k] = 10 \* k;

printf("\n %d", intArray[k]);

}// end for

pointerArray( startPtr, endPtr); // giving the addresses

puts("\n");

}// end main

void pointerArray(int \*Fun\_startPtr, int \*Fun\_endPtr)

{

while (Fun\_startPtr < Fun\_endPtr)

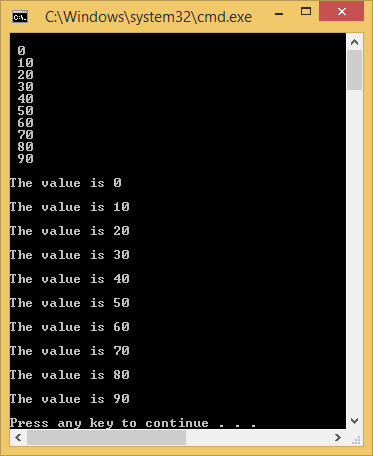
{

printf("\n\nThe value is %d", \*Fun\_startPtr);

Fun\_startPtr++;

} // end while

}// end user



//

#include <stdio.h>

int \* pointerArray(int \*Fun\_startPtr, int \*Fun\_endPtr, int searchFor); // starting address and ending address of array

void main(void)

{

int intArray[10] = { 0 };

int \*startPtr = &intArray[0]; // starting address of array

int \*endPtr = &intArray[9]; // ending address of array

int search4 = 30;

size\_t k;

for (k = 0; k < 10; k++)

{

intArray[k] = 10 \* k;

printf("\n %2d located at %p", intArray[k], &intArray[k]);

}// end for

printf("\n\n%p \n\n", pointerArray(startPtr, endPtr, search4)); // giving the addresses

puts("\n");

}// end main

int \* pointerArray(int \*Fun\_startPtr, int \*Fun\_endPtr, int searchFor)

{

while (Fun\_startPtr <= Fun\_endPtr)

{

printf("\n\nThe value is %d", \*Fun\_startPtr);

if (searchFor == \*Fun\_startPtr)

{

return Fun\_startPtr;

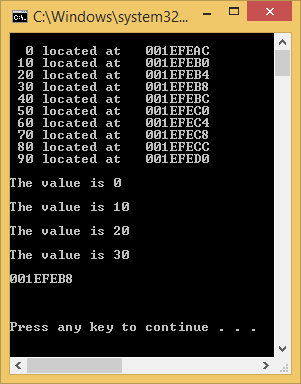
}// end if

Fun\_startPtr++;

} // end while

return NULL;

}// end user



//

#include <stdio.h>

void circleMath(double Fun\_radius, double \*diameter, double \*area);

void main(void)

{

double radius = -1.0;

double diameter = 0.0;

double area = 0.0;

puts("\nEnter a radius:");

scanf\_s("%lf", &radius);

circleMath(radius, &diameter, &area);

printf("\nThe diameter is %.8lf \n", diameter);

printf("\nThe area is %.8lf \n", area);

}//end main

void circleMath(double Fun\_radius, double \*diameter, double \*area)

{

double myPi = 3.14159265;

\*diameter = 2.0 \* Fun\_radius;

\*area = myPi \* Fun\_radius \* Fun\_radius;

}// end user

