Quiz 5 Name:________________________

Question 1. (10pt) Given:

```
void fFunc(int *ptr, int size);     // ptr to an array with size
void gFunc(int array[], int size);  // array with size
void hFunc(int);
void jFunc(int&);
void kFunc(int*);

int main() {
     int myArray[5] = {1, 2, 3, 4, 5};
     int *ptr;

     // Your code comes here
}
```

a. (1pt) Please show how to call fFunc() with myArray as argument.

fFunc(myArray, 5)

b. (1pt) Please show how to call gFunc() with myArray as argument.

gFunc(myArray, 5)

c. (2pt) Please show how to call hFunc() with index-0 element of myArray as argument.

hFunc(myArray[0])

d. (3pt) Please show how to call jFunc() with index-1 element of myArray as argument.

jFunc(myArray[1])

e. (3pt) Please show how to call kFunc() with index-2 element of myArray as argument.

kFunc(&(myArray[2]))

Question 3. (20pt)

a. (2pt) Write a C++ function to receive an integer and return a boolean indicating if the integer is an even number.

```
bool IsEven(int n) {
    return ((n % 2) == 0);
}
```

b. (4pt) Please show the implementation of the following function:

```
int ExtractEven(int input[], int size, int output[])
   // Function goes through input[], copy and return even numbers in output[]
   // size: size of the input array
   // returns: size of the output array
   // Pre-condition: output[] is allocated with size-number of elements
   // For example:
   //       input = {1, 2, 2, 3, 4, 5}   size=6
   // When return:
   //       output = {2, 2, 4}, return value 3
```

```
int ReturnEven(int input[], int size, int output[]) {
    int out_size = 0;
    for (int i = 0; i<size; i++) {
        if (IsEven(input[i])) {
            output[out_size++] = input[i];
        }
    }
    return out_size;
}
```

c. (6pt) Please show the implementation of the following function:

```
int KeepEvenAndCompact(int input[], int size)
   // Function goes through input[], remove odd numbers, the resulting input[]
   //   must be contain only even numbers found in the same order as the input[]
   // size: size of the input array
   // returns: size of the resulting array
   // For example:
   //        input = {1, 2, 2, 3, 4, 5}   size=6
   // When return:
   //        input = {2, 2, 4}, return value 3
```

```
int KeepEvenAndCompact(int input[], int size) {
    int returnSize = 0;
    for (int i = 0; i<size; i++) {
        if (IsEven(input[i])) {
            input[returnSize] = input[i];
            returnSize++;
        }
    }
    return returnSize;
}
```

d. (8pt) Please show the implementation of the following function:

```
int KeepEvenCompactAndUnique(int input[], int size)
   // Function goes through input[], remove odd numbers and any duplicates,
   // the resulting input[] contains only unique even numbers found in the
   // same order as the input[]
   // size: size of the input array
   // returns: size of the resulting array
   // For example:
   //       input = {1, 2, 2, 3, 4, 5}   size=6
   // When return:
   //       input = {2, 4}, return value 2
```

```
int RemoveCompactAndUnique(int input[], int size) {
    int returnSize = 1;
    for (int i = 1; i<size; i++) {
        if (IsEven(input[i])) {
            bool dup = false;
            int check = 0;
            while ((!dup) && (check < returnSize)) {
                dup = (input[check] == input[i]);
                check++;
            }
            if (!dup) { // unique odd number ...
                input[returnSize] = input[i];
                returnSize++;
            }
        }
    }
    return returnSize;
}
```