PROGRAM 1:

```
//Toni Hunter 187009925
//FINAL Q1
//
//Two functions: average, mode with 1 dimensional array of integers and the size of the array as parameters.
//Average should return number as float, Mode should return int.

#include <stdio.h>


//prototypes
float average(int array[], int num);
int mode(int array[], int num);

int main(void)
{
   int num; // number of elements, user input

   printf("Please input the number of elements to be stored in the array: ");
   scanf("%d", &num);

   printf("input %d elements into the array: \n", num);
   printf("\n");

   int array[num]; //initialize to user input size, must be after scanf to work
   for (int i = 0; i < num; i++) //starts at index zero until num is reached
   {
      printf("element %d: ", i);
      scanf("%d", &array[i]); //set element at location i (starts at 0) to i
   }


    /*printf("\nEntered elements: \n");
    for (int i = 0; i < num; i++)
    {
       printf("%d \n", array[i]);
    }*/

    printf("\nAverage: %.1f", average(array, num)); //function call average
    printf("\nMode: %d", mode(array, num)); //function call num

}

float average(int array[], int num)
{
   int total = 0;

   // loop through the elements
   for ( int i = 0; i < num; i++ )
   {
      total += array[i]; // add elements in array
   }
   float avg = total / num; // calculates the average
```

```
    return avg; //returns double average
}

int mode(int array[], int num)
{
    int maxValue = 0;
    int maxCount = 0;

    for (int i = 0; i < num; ++i)
    {
        int count = 0; //initializes count to 0 for the array
        for (int j = 0; j < num; ++j)
        {
            if (array[j] == array[i])
            count++; //increments count for each element
        }
        if (count > maxCount)
        {
            maxCount = count; // sets maxcount to count
            maxValue = array[i]; // sets max element
       }
   }

   return maxValue;
}
```

PROGRAM 2:

```
//Toni Hunter 187009925
//FINAL Q2
//
//count the frequency of elements in an array of any size(greater than zero).

#include <stdio.h>
#include <stdlib.h>

int main()
{
    int num; //number of elements in the array, user input

    printf("Please enter number of elements to be stored into the array: ");
    scanf("%d", &num);

    while (num > 0)
    {
        int array[num]; //initialize to user input size, must be after scanf to work
        for (int i = 0; i < num; i++) //starts at index zero until num is reached
        {
            printf("element %d: ", i);
            scanf("%d", &array[i]); //set element at location i (starts at 0) to i
        }
```

```
        printf("the frequency of all the elements of the array: \n");



        for (int i = 0; i < num; ++i)
        {
            int count = 0; //initializes count to 0 for the array
            for (int j = 0; j < num; ++j)
            {
                if (array[j] == array[i])
                {
                    count++; //increments count for each element
                }
            }
            printf("%d occurs %d times.\n", array[i], count);
        }

    }
    printf("Please enter an array size larger than 0!");
}
```

PROGRAM 3:

```
//Toni Hunter 187009925
//FINAL Q3 UNFINISHED!!!
//
//Delete or insert a value into an array (sorted, 7 elements max or smaller).

#include <stdio.h>
#include <stdlib.h>

int main()
{
    int array[7]; // array with max size 7
    int num; //array size, user input
    char user;
    int val;

    printf("Please input the number of elements to be stored in the array: ");
    scanf("%d", &num);

    printf("input %d elements into the array: \n", num);
    printf("\n");

    while (num > 0 && num <= 7)
    {
        int array[num]; //initialize to user input size, must be after scanf to work
        for (int i = 0; i < num; i++) //starts at index zero until num is reached
        {
            printf("element %d: ", i);
            scanf("%d", &array[i]); //set element at location i (starts at 0) to i
        }
```

```
        printf("DELETE(D) or INSERT(I) or EXIT(E): ");
        scanf("%c", &user);

        if (user == 'D' || 'd')
        {
            printf("input value to be deleted: ");
            scanf("%d", val);

            printf("\nThe existing array is: \n");
            for (int i = 0; i < num; i++)
            {
                printf("%d ", array[i]);
            }

            printf("The array after the delete is: \n");

            for (int i = 0; i < num; i++)
            {
                if (array[i] == val)
                {
                    array[i] = array[i + 1];
                }
                num--;
                printf("%d ", array[i]);
            }


        }

        /*else if (user == 'I' || 'i')
        {

        }
        else if (user == 'E' || 'e')
        {
            break;
        }*/


    }

}
```

PROGRAM 4:

```
//Toni Hunter 187009925
//FINAL Q4
//
//Ask info from 3 students, store the information into an array of structures.
//info: student ID, last name, programming grade, math grade.
//returns average grade for programming class, math class, max grade from math and min grade from math.
//
#include <stdio.h>
```

```
struct student {
    //members in struct student
    char FirstName[20]; //can have up to 20 chars
    char LastName[20];
    char StudentID[20];
    float Pro;
    float Math;

} s[3]; //student info array containing all members

int main() {
    int i;
    float Avg_Pro=0;
    float Avg_Math=0;
    float Max_Math=0;
    float Min_Math=0;
    printf("Please enter student information:\n");

    // storing information for each student in s array, starting at 0
    for (i = 0; i < 3; ++i)
    {
        printf("\n");
        printf("Enter first name: ");
        scanf("%s", s[i].FirstName);
        printf("Enter Last name: ");
        scanf("%s", s[i].LastName);
        printf("Enter grade for programming: ");
        scanf("%f", &s[i].Pro);
        printf("Enter grade for Math: ");
        scanf("%f", &s[i].Math);

    }
    printf("\nBelow is the information of 3 students:");
    printf("\n");

    // displays information on each student starting at 0
    for (i = 0; i < 3; ++i)
    {
        //calculating averages
        Avg_Pro = (Avg_Pro + s[i].Pro)/2;
        Avg_Math = (Avg_Math +  s[i].Math)/2;

        printf("First name: ");
        puts(s[i].FirstName);

        printf("Last name: ");
        puts(s[i].LastName);

        printf("grade for programming: %.1f\n", s[i].Pro);
        printf("grade for Math: %.1f", s[i].Math);
        printf("\n");
        printf("\n");

        if (Min_Math > s[i].Math)
```

```
        {
            Min_Math = s[i].Math;
        }

        if (Max_Math < s[i].Math)
        {
            Max_Math = s[i].Math;
        }

    }

    printf("PROGRAMMING AVERAGE: %.1f\n", Avg_Pro);
    printf("MATH AVERAGE: %.1f\n", Avg_Math);
    printf("MAXIMUM MATH GRADE: %.1f\n", Max_Math);
    printf("MINIMUM MATH GRADE: %.1f\n", Min_Math);
    return 0;
}
```