

Name (PRINTED): _____

Student ID #: _____

Section # (or TA's: _____
name and time)

CMSC 106

Quiz #8

Fall 2007

Write the code as requested - you may assume there is a # defined constant named SIZE that holds the value referred to and you may assume that one section of code you write can be used by other sections.

1. [8 pts.] Write a function that takes an array of size SIZE through the parameter then requests and reads the integers typed until either the array is filled or the user enters a 0. The number of values typed before the 0 or until the array was filled should also be passed back to the function who called this one through a parameter. The return value of the function will be 0 if the array is completely full or it will return a 1 if there is one or more empty slots in the array.

```
int roomLeftFill(int arr[], int *numFilled){
    int count = 0; int currValue = 0;
    scanf("%d", &currValue);
    while(currValue != 0 && count < SIZE){
        arr[count] = currValue;
        count++;
        if(count < SIZE){
            scanf("%d", &currValue);
        }
    }
    *numFilled = count;
    if (*numFilled == SIZE){
        return 0;
    } else {
        return 1;
    }
}
```

2. [8 pts.] Write a function that takes in an integer array of size SIZE that is already filled with some values. It also takes an integer parameter to tell how many of the array elements are filled (they are filled from the beginning of the array to that point). It then returns both the average and the highest value (the two things people ask about exam scores).

```
void aveHigh(int arr[], int numFilled, float *ave, int *high){
    float sum = arr[0];
    int count = 0;
    int highSoFar = arr[0];
    for(count = 1; count < numFilled; count++){
        sum += arr[count];
        if (arr[count] > highSoFar){
            highSoFar = arr[count];
        }
    }
    *ave = sum/numFilled;
    *high = highSoFar;
}
```

3. [15 pts.] Write a main function that will create a SIZE element array of integers, fill that array that array using one of the functions written on the other side of this paper and print the average and highest values (in sentences) using the other function written on the other side of this paper.

```
int main(){
    float ave;
    int high, usedSize;
    int values[SIZE];
    int isNotFull = roomLeftFill(values, &usedSize);
    if (isNotFull){
        printf("array not full\n");
    } else {
        printf("array full\n");
    }
    aveHigh(values, usedSize, &ave, &high);
    printf("the average = %.2f\n", ave);
    printf("the high = %d\n", high);
    return 0;
}
```