//

// main.c

// Chpt7PP1

//

// Created by Randy McMillan on 10/26/13.

// Copyright (c) 2013 Randy McMillan. All rights reserved.

//

//1. Write a C program that lets a user enter up to 15 numbers (it will stop accepting numbers either because of a sentinel value or because it reaches the limit of 15 numbers). Display the entered numbers in reverse order (the opposite order in which they were entered).

#include <stdio.h>

#define SENTINEL = -1;

double array[15];

void myMain();

void printArray();

int main(int argc, const char \* argv[])

{

myMain();

return 0;

}

void myMain(){

for (int i=0; i<=14; i++) {

printf("Let array[%i] in the array = ",i);

scanf("%lf",&array[i]);

if (array[i] == -1) {

i=15;//force control

//printArray();

}

}//end for

printArray();

}

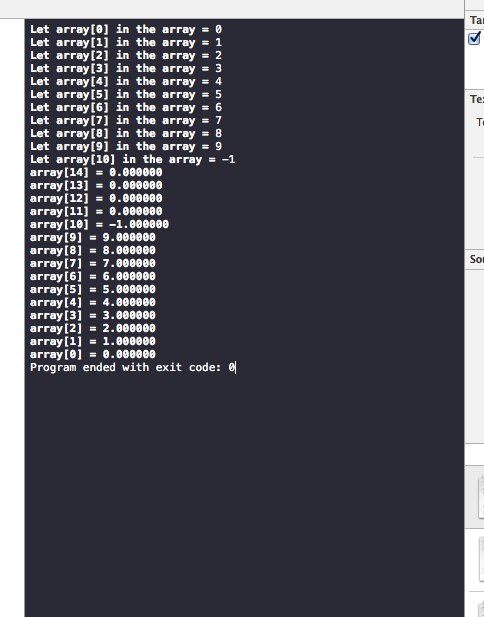
void printArray(){

for (int i=14; i>=0; i--) {

printf("array[%i] = %lf\n",i,array[i]);

}//end for

}



//

// main.c

// Chpt7PP2

//

// Created by Randy McMillan on 10/26/13.

// Copyright (c) 2013 Randy McMillan. All rights reserved.

//

//2. Write a C program that fills an array with 100 random\* numbers between 0 and 1. Then, go through the array and count how many numbers are <0.5 and how many are >= 0.5. Display the contents of the array and the two count totals.

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#include <math.h>

void myMain();

void populateArray();

float returnRand();

float returnClock();

float array[100];

int greaterThan05,lessThan05;

void calcCountTotals();

void displayCountTotals();

int main(int argc, const char \* argv[])

{

myMain();

return 0;

}

void myMain(){

populateArray();

calcCountTotals();

displayCountTotals();

}

void populateArray(){

for (int i=0; i<100; i++) {

//we know this isn't true random

//array[i]=rand();

array[i]=returnRand();

printf("Value at array[%i] = %0.1lf\n",i,array[i]);

}

}

float returnRand(){

double clock = returnClock();

clock = clock/10;

int intpart = (int)clock;

double decpart = clock - intpart;

return decpart;

}

float returnClock(){

return clock();

}

void calcCountTotals(){

for (int i=0; i< 100; i++) {

if (array[i]< 0.5) {

lessThan05++;

} else {

greaterThan05++;

}

}

}

void displayCountTotals(){

printf("\n %i values were greater than or equal to 0.5\n",greaterThan05);

printf("\n %i values were less than 0.5\n\n",lessThan05);

//The average values approach 50 and 50 when the program is run repeatedly which suggests a "standard distribution" between 0-1 with a bias toward "greater than 0.5" because 0.5 is included in the "greater than range".

}

