Barak Barclay

Leslie Tekamp

ECE-1021

23 September 2015

HWK-3

//

// preprocessor directives

// start main

// init variable

// while loop (num >= 0)

// prompt user to enter number

// read in num

// if (num ==48)

// print 0 to screen

// end if

// copy and paste if statment changing 48 and 0 to correct numbers

// end while

// end main

//

#include <stdio.h>

int main(void)

{

int num = 0;

while (num >= 0)

{

printf\_s("\nEnter a integer that's greater than or equal to 48, but less than or equal to 57.\n");

scanf\_s("%d", &num);

if (num == 48)

{

printf\_s("0");

}

if (num == 49)

{

printf\_s("1");

}

if (num == 50)

{

printf\_s("2");

}

if (num == 51)

{

printf\_s("3");

}

if (num == 52)

{

printf\_s("4");

}

if (num == 53)

{

printf\_s("5");

}

if (num == 54)

{

printf\_s("6");

}

if (num == 55)

{

printf\_s("7");

}

if (num == 56)

{

printf\_s("8");

}

if (num == 57)

{

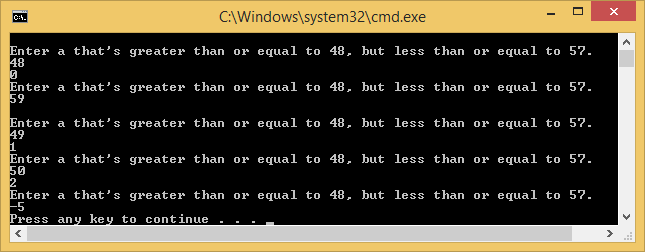
printf\_s("9");

}

}

return 0;

}



//

// HWK3(2).cpp

//

// By: Barak Barclay

// Date: 23 Sept 2015

//

// Problem Statement:Write a program which prompts the user to enter in two double numbers and one

// integer number. Read in the numbers. Calculate the sum and product of the three numbers and

// print the results to the screen. Use explicit type casting.

//

// %%%% Algorthim %%%%

//

// preprocessor directives

// start main

// init variables

// prompt user to enter double number

// read in d1

// prompt user to enter double number

// read in d2

// prompt user to enter integer number

// read in num

// do sum and product of the 3 numbers

// print sum and product

// end main

//

#include <stdio.h>

int main(void)

{

double d1 = 0;

double d2 = 0;

int num = 0;

double sum = 0;

double product = 0;

printf\_s("Enter a double number:");

scanf\_s("%lf", &d1);

printf\_s("Enter another double number:");

scanf\_s("%lf", &d2);

printf\_s("Enter an integer number:");

scanf\_s("%d", &num);

sum = (double) (d1 + d2 + num);

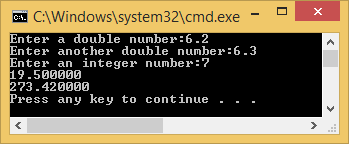
product = (double) (d1 \* d2 \* num);

printf\_s("%lf\n", sum);

printf\_s("%lf\n", product);

return 0;

}



//

// HWK3(3).cpp

//

// By: Barak Barclay

// Date: 23 Sept 2015

//

// Problem Statement:Using a while loop, write a program which prompts the user to enter in integer

// numbers (between 0 and 9 - inclusive) to average together. Error check each number entered,

// reject the number if it is not between 0 and 9 and write a message to the screen that the number

// entered was out of bounds. After the user enters in -99, error check to make sure the user entered

// in at least one valid number, if not - print an error message, else print the average value and the

// remainder value to the screen.

//

// %%%% Algorthim %%%%

//

// preprocessor directives

// start main

// init variable

// while loop (num != -99)

// prompt user to enter a integer that's greater than or equal to 0, but less than or equal to 9

// read in num

// increment i

// if (num != -99 && (num <= 0 || num >= 9))

// set num = 0

// print the number entered was out of bounds

// decrement i

// end if

// else

// add num to total

// end else

// end while

// print average

// print remainder

// end main

//

#include <stdio.h>

int main(void)

{

int num = 0;

int total = 0;

int i = 0;

while (num != -99)

{

printf\_s("Enter a integer that's greater than or equal to 0, but less than or equal to 9.\n");

scanf\_s("%d", &num);

i++;

if (num != -99 && (num <= 0 || num >= 9))

{

num = 0;

printf\_s("The number entered was out of bounds.\n");

i--;

}

else

{

total += num;

}

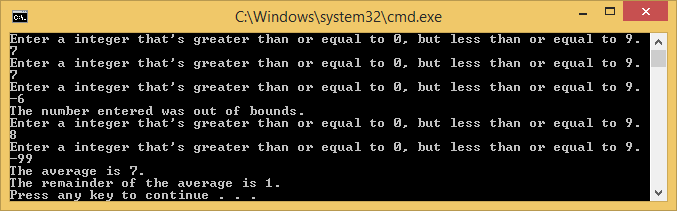
} //end while

printf\_s("The average is %d.\n", (total + 99) / (i - 1));

printf\_s("The remainder of the average is %d.\n", (total + 99) % (i - 1));

return 0;

}//main



//

// HWK3(4).cpp

//

// By: Barak Barclay

// Date: 23 Sept 2015

//

// Problem Statement:Write a program which prompts the user to enter in a float number. Read in the number.

// Convert the float to an integer number. If the number is odd, print it and the next 9 odd numbers to

// the screen. If the number is even, print the next 9 even numbers to the screen

//

// %%%% Algorthim %%%%

//

// preprocessor directives

// start main

// init variables

// while loop (true)

// prompt user to enter a float number

// read in f

// set i = f

// if (i & 1)

// print next 9 odd numbers

// end if

// else

// print next 9 even numbers

// end else

// end while

// end main

//

#include<stdio.h>

int main(void)

{

while (true)

{

float f = 0;

int i = 0;

printf\_s("Enter a float number:");

scanf\_s("%f", &f);

i = f;

if (i & 1)

printf("%d, %d, %d, %d, %d, %d, %d, %d, %d\n", i + 2, i + 4, i + 6, i + 8, i + 10, i + 12, i + 14, i + 16, i + 18);

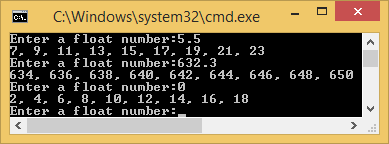
else

printf("%d, %d, %d, %d, %d, %d, %d, %d, %d\n", i + 2, i + 4, i + 6, i + 8, i + 10, i + 12, i + 14, i + 16, i + 18);

}

return 0;

}



//

// HWK3(5).cpp

//

// By: Barak Barclay

// Date: 23 Sept 2015

//

// Problem Statement:Money and float math. Prompt the user to enter in the cost of an item.

// Assume the sales tax is 6%. Calculate the total amount due. Work this problem using floats

// for the math operations. Work it again using integers, convert by multiplying dollar/cent amount

// by 100 – i.e $33.33 \* 100 = 3333 (an int). Final step is to convert back to float and divide by 100.

//

// %%%% Algorthim %%%%

//

// preprocessor directives

// start main

// init variables

// while loop (true)

// prompt user to enter the cost of an item

// read in f

// ftot = 1.06 \* f;

// set f2 = f \* 10000;

// set i = f2 + f2 \* 6 / 100;

// set itot = i / 10000;

// print ftot

// print itot

// end while

// end main

//

#include<stdio.h>

int main(void)

{

while (true)

{

float f = 0;

float f2 = 0;

float ftot = 0;

int i = 0;

float itot = 0;

printf\_s("Enter the cost of an item:");

scanf\_s("%f", &f);

ftot = 1.06 \* f;

f2 = f \* 10000;

i = f2 + f2 \* 6 / 100;

itot = (float) i / 10000;

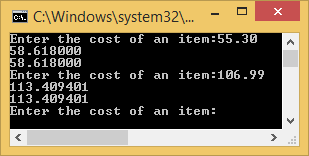
printf("%f\n", ftot);

printf("%f\n", itot);

}

return 0;

}



1. Kjhjh

//

// HWK3(6part1).cpp

//

// By: Barak Barclay

// Date: 23 Sept 2015

//

// Problem Statement: Work book problem 4.16 page 153. Do each problem using multiple puts

// or printf statements. Do them again using a single puts or printf statement inside a loop.

//

// %%%% Algorthim %%%%

//

// preprocessor directives

// start main

// multiple printf statements for patterns

// end main

//

#include<stdio.h>

int main(void)

{

printf\_s("\*\n");

printf\_s("\*\*\n");

printf\_s("\*\*\*\n");

printf\_s("\*\*\*\*\n");

printf\_s("\*\*\*\*\*\n");

printf\_s("\*\*\*\*\*\*\n");

printf\_s("\*\*\*\*\*\*\*\n");

printf\_s("\*\*\*\*\*\*\*\*\n");

printf\_s("\*\*\*\*\*\*\*\*\*\n");

printf\_s("\*\*\*\*\*\*\*\*\*\*\n");

printf\_s("\n");

printf\_s("\*\*\*\*\*\*\*\*\*\*\n");

printf\_s("\*\*\*\*\*\*\*\*\*\n");

printf\_s("\*\*\*\*\*\*\*\*\n");

printf\_s("\*\*\*\*\*\*\*\n");

printf\_s("\*\*\*\*\*\*\n");

printf\_s("\*\*\*\*\*\n");

printf\_s("\*\*\*\*\n");

printf\_s("\*\*\*\n");

printf\_s("\*\*\n");

printf\_s("\*\n");

printf\_s("\n");

printf\_s("\*\*\*\*\*\*\*\*\*\*\n");

printf\_s(" \*\*\*\*\*\*\*\*\*\n");

printf\_s(" \*\*\*\*\*\*\*\*\n");

printf\_s(" \*\*\*\*\*\*\*\n");

printf\_s(" \*\*\*\*\*\*\n");

printf\_s(" \*\*\*\*\*\n");

printf\_s(" \*\*\*\*\n");

printf\_s(" \*\*\*\n");

printf\_s(" \*\*\n");

printf\_s(" \*\n");

printf\_s("\n");

printf\_s(" \*\n");

printf\_s(" \*\*\n");

printf\_s(" \*\*\*\n");

printf\_s(" \*\*\*\*\n");

printf\_s(" \*\*\*\*\*\n");

printf\_s(" \*\*\*\*\*\*\n");

printf\_s(" \*\*\*\*\*\*\*\n");

printf\_s(" \*\*\*\*\*\*\*\*\n");

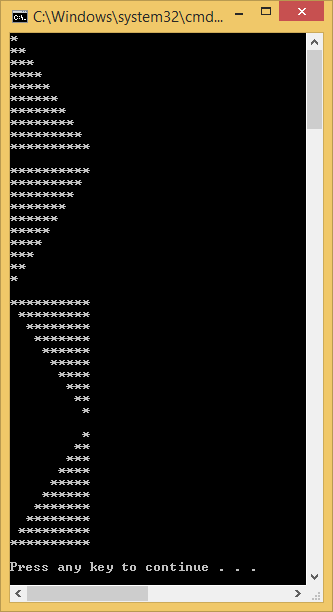
printf\_s(" \*\*\*\*\*\*\*\*\*\n");

printf\_s("\*\*\*\*\*\*\*\*\*\*\n");

printf\_s("\n");

return 0;

}



//

// HWK3(6part2).cpp

//

// By: Barak Barclay

// Date: 23 Sept 2015

//

// Problem Statement: Work book problem 4.16 page 153. Do each problem using multiple puts

// or printf statements. Do them again using a single puts or printf statement inside a loop.

//

// %%%% Algorthim %%%%

//

// preprocessor directives

// start main

// init variables

// for loops to create patterns

// end main

//

#include<stdio.h>

int main(void)

{

int i;

int i1;

int i2 = 10;

for (i = 1; i <= 10; i++)

{

for (i1 = 1; i1 <= i; i1++)

{

printf\_s("\*");

}

printf\_s("\n");

}

printf\_s("\n");

for (i = 1; i <= 10; i++)

{

for (i1 = i; i1 <= 10; i1++)

{

printf\_s("\*");

}

printf\_s("\n");

}

printf\_s("\n");

i2 = 10;

for (i = 1; i <= 10; i++)

{

for (i1 = 10 - i2; i1 >= 1; i1--)

{

printf\_s(" ");

}

i2--;

for (i1 = 10; i1 >= i; i1--)

{

printf\_s("\*");

}

printf\_s("\n");

}

printf\_s("\n");

for (i = 1; i <= 10; i++)

{

for (i1 = 1; i1 <= (10 - i); i1++)

{

printf\_s(" ");

}

for (i1 = 1; i1 <= i; i1++)

{

printf\_s("\*");

}

printf\_s("\n");

}

printf\_s("\n");

return 0;

}

