/\*

In Class Challenge 1

This program displays a 9x9 sudoku table

Uses a whole bunch of printf statements

Author Peter Dobbs

Created: September 2015

\*/

#include <stdio.h>

#include <stdlib.h>

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

printf("++=====================================++\n");

printf("|| 2 | | || | | 3 || 8 | | ||\n");

printf("||-------------------------------------||\n");

printf("|| | 6 | || 7 | 2 | || | 1 | ||\n");

printf("||-------------------------------------||\n");

printf("|| 7 | | 8 || | | 9 || 4 | | ||\n");

printf("||=====================================||\n");

printf("|| | 8 | || | 6 | 4 || 7 | | ||\n");

printf("||-------------------------------------||\n");

printf("|| 5 | | 6 || 1 | | 7 || 2 | | 8 ||\n");

printf("||-------------------------------------||\n");

printf("|| | | 7 || 5 | 8 | || | 9 | ||\n");

printf("||=====================================||\n");

printf("|| | | 3 || 9 | | || 1 | | 7 ||\n");

printf("||-------------------------------------||\n");

printf("|| | 1 | || | 7 | 6 || | 8 | ||\n");

printf("||-------------------------------------||\n");

printf("|| | | 5 || 8 | | || | | 4 ||\n");

printf("++=====================================++\n");

system("PAUSE");

return 0;

}

/\*

In Class Challenge 2

This program prints three different phone numbers

Uses a whole bunch of printf statements

Author Peter Dobbs

Created: September 2015

\*/

#include <stdio.h>

#include <stdlib.h>

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

//variable declaration

char \*phoneNum1 = "(123) 456-7890";

char \*phoneNum2 = "(098) 765-4321";

printf("%s\n", phoneNum1);

printf("%s\n", phoneNum2);

//phoneNum1

int areaCode1, prefix1, suffix1;

areaCode1 = 414;

prefix1 = 774;

suffix1 = 9878;

//phoneNum2

int areaCode2, prefix2, suffix2;

areaCode2 = 414;

prefix2 = 429;

suffix2 = 6622;

//phoneNum1

printf("(%d", areaCode1);

printf(") %d", prefix1);

printf("-%d\n", suffix1);

//phoneNum2

printf("(%d", areaCode2);

printf(") %d", prefix2);

printf("-%d\n", suffix2);

//phoneNum3

int areaCode3, prefix3, suffix3;

areaCode2 = 8;

prefix2 = 99;

suffix2 = 199;

//phoneNum3

printf("(60%d", areaCode3); //accounts for leading zeros

printf(") 9%d", prefix3);

printf("-0%d\n", suffix3); //accounts for leading zeros

system("PAUSE");

return 0;

}

/\*

In Class Challenge 3

Robodialing:

Uses a loop to change phone number

Author Peter Dobbs

Created: 9/16/15

Updated: 9/21/15

\*/

#include <stdio.h>

#include <stdlib.h>

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

//robodialing

//starting number: (414) 555-0110

//increment the phone number by one until you reach (414) 555-0111

//(phoneArea) phonePref-phoneSuff

int phoneArea = 414;

int phonePref = 555;

int phoneSuff = 9000;

int i = 0;

// for (i=0; i<10; i++){

//// printf("(%d) ", phoneArea);

//// printf("%d-", phonePref);

//// printf("%04d\n", phoneSuff);

// if (phoneSuff <= 999){

// printf("(%d) %d-%04d\n", phoneArea, phonePref, phoneSuff+i);

// }

// if (phoneSuff > 999){

// printf("(%d) %d-%d\n", phoneArea, phonePref, phoneSuff+i);

// }

// }

for(i=0; i<1000; i++){

printf("(%d) %d-%d\n", phoneArea, phonePref, phoneSuff+i);

}

return 0;

}

/\*

In Class Challenge 4

This program calculates the sum of numbers

between two given integers and the product

of those numbers, then finds the ratio of

the sum and the product. The range should be

exclusive

Author Peter Dobbs

Created 9/23/15

\*/

#include <stdio.h>

#include <stdlib.h>

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

//variable declaration and initialization

int a = 10; //given integer 1

int b = 20; //given integer 2

int i = 0;

int range\_var = b-(a+1); //variable for range

int range\_array[range\_var]; //array holding the range

float sum = 0;

float prod = 1;

float ratio = 0; //quotient of sum and product

//print the range

printf("Range = %i\n", range\_var);

for(i = 0; i<range\_var; i++){

range\_array[i] = a+1;

a++;

}

//sum of range(a to b)

for(i = 0; i<range\_var; i++){

sum = sum + range\_array[i];

}

printf("Sum: %g\n", sum);

//product of range(a to b)

for(i = 0; i<range\_var; i++){

prod = prod \* range\_array[i];

}

printf("Product: %g\n", prod);

//calculate (sum of range(a to b))/(product of range(a to b))

ratio = sum/prod;

printf("Ratio: %g\n", ratio);

system("PAUSE");

return 0;

}

/\*

In Class Challenge 5

This program robodials with user input

Peter Dobbs

9/28/2015

\*/

#include <stdio.h>

#include <stdlib.h>

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

//increment the phone number by one until you reach (414) 555-0111

//variable declaration //initialize to zero

int phoneArea=0; //variable for areacode

int phonePref=0; //variable for exchange/prefix

int phoneSuff=0; //variable for suffix

int i = 0;

//user input

printf("Enter the area code:\n");

scanf("%i", &phoneArea); //takes input and assigns it to 'phoneArea'

printf("Enter the exchange/prefix:\n");

scanf("%i", &phonePref); //takes input and assigns it to 'phonePref'

printf("Enter the suffix:\n");

scanf("%i", &phoneSuff); //takes input and assigns it to 'phoneSuff'

//prints phone list to console

for (i=0; i<10; i++){

if (phoneSuff < 1000){

printf("(%d) %d-%04d\n", phoneArea, phonePref, phoneSuff+i);

}

if (phoneSuff >= 1000){

printf("(%d) %d-%d\n", phoneArea, phonePref, phoneSuff+i);

}

}

return 0;

}

/\*

In Class Challenge 6

This program takes the input of a text file and

counts how many times the number 5 is listed.

It only works for files with less than 5000 numbers.

Peter N. Dobbs

10/5/2015

\*/

#include <stdio.h>

#include <stdlib.h>

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

// local variable declaration and initialization

float a[5000]; //array with n elements

float sum = 0; //to be used for sum of array elements

float avg = 0; //to be used for average of array elements

int i; //to be used as index for the for-loops

int count; //to be used for the number of integer usages

int num = 0;

FILE \*fp; //the star(\*) indicates the pointer.

// open the file

fp = fopen("in.txt", "r"); // 'r': read, 'w': write, 'a': append

if (fp==NULL) { // error catch

printf("File not found\n");

return 1; // ends the program

}

i=0;

while(fscanf(fp, "%g",&a[i])!=EOF){

i++;

}

num = i;

// close the file

fclose(fp);

count = 0;

for (i=0; i<num; i++){

if (a[i]==5) count++;

}

printf("5 is listed %i times\n", count); //should be 106 times

system("PAUSE");

return 0;

}

/\*

In Class Challenge 7

This program pulls data from text files to do a comparison

This program only works for files < 50 points.

If you run more than once, the text in the file will be

duplicated within the text file.

Author Peter N. Dobbs

Created: 10/12/2015

\*/

#include <stdio.h>

#include <stdlib.h>

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

// local variable declaration and initialization

char first\_name[100];

char second\_name[100];

int catNum[50];

char patientStatus;

int i = 0;

FILE \*fpName; //the star(\*) indicates the pointer.

FILE \*fpNumb;

FILE \*fpOut;

// open the files, 'r': read, 'w': write, 'a': append

fpName = fopen("med\_data\_1.txt", "r"); // reads from file

fpNumb = fopen("med\_data\_2.txt", "r"); // reads from file

fpOut = fopen("output.txt", "a"); // appends to file

if (fpName==NULL) { // error catch

printf("File 1 not found\n");

return 1; // ends the program

}

if (fpNumb==NULL) { // error catch

printf("File 2 not found\n");

return 2; // ends the program

}

for(i=0; i<50; i++){

// read from files

fscanf(fpName,"%s %s",&first\_name[0],&second\_name[0]);

fscanf(fpNumb,"%d",&catNum[i]);

// write to file

//printf("%d\t", catNum[i]);

fprintf(fpOut, "%d\t", catNum[i]);

if(catNum[i]<=5000) fprintf(fpOut, "normal\t\t");//printf("normal\t\t");

else fprintf(fpOut, "emergency\t"); //printf("emergency\t");

//printf("%s %s \n", first\_name, second\_name);

fprintf(fpOut, "%s %s \n", first\_name, second\_name);

}

fclose(fpName);

fclose(fpNumb);

fclose(fpOut);

return 0;

}

/\*

avg1

This Program calculates the average of 3 hardcoded numbers

Uses float variables, arithmetic, printf

Author Peter Dobbs

Created 10/10/15

\*/

#include <stdio.h>

#include <stdlib.h>

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

// declare variables

float a;

float b;

float c;

float avg;

// assigning values

a = 3;

b = 5;

c = 8;

// calculating average

avg = (a+b+c)/3.0; //should be 5.3333333

// print the result

printf("The average of our three numbers is %f\n",avg);

return 0;

}

/\*

avg1challenge

This program calculates the avg of 4 numbers

Uses float variables, arithmetic, and printf

@author Peter N Dobbs

Created: 10/10/2015

\*/

#include <stdio.h>

#include <stdlib.h>

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

// declare variables

float a;

float b;

float c;

float d;

float avg;

// assigning values

a = 3;

b = 5;

c = 8;

d = 4;

// calculating average

avg = (a+b+c+d)/4.0;

// print the result

printf("The average of our three numbers is %f\n",avg);

return 0;

}

/\*

avg2 - from Lesson 1c

This program calculates the average of 3 numbers

Uses array, arithmetic, printf

@author Peter N Dobbs

Created: 10/10/2015

\*/

#include <stdio.h>

#include <stdlib.h>

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

// declare variables

float a[3]; //a is a 3 element array

float avg;

// assigning values

a[0] = 3;

a[1] = 5;

a[2] = 8;

// calculating average

avg = (a[0]+a[1]+a[2])/3.0;

// print the result

printf("The average of our three numbers is %f\n",avg);

return 0;

}

/\*

avg3 - from Lesson 1c

This program calculates the average of 3 numbers,

Uses an array and a loop.

@author Peter N Dobbs

Created: 10/10/2015

\*/

#include <stdio.h>

#include <stdlib.h>

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

// declare variables

float a[3]; //a is a 3 element array

float sum;

float avg;

int i;

// assigning values

a[0] = 3;

a[1] = 5;

a[2] = 8;

//new algorithm

/\*

sum = 0;

i = 0;

while(i<3){

sum = sum + a[i];

//to print int use d (decimal), to print float use f (float)

printf("%d %f\n", i, sum);

i++; // much better than i=i+1

}

\*/

sum = 0;

for(i=0;i<3;i++){

sum = sum + a[i];

}

avg = sum/3;

// print the result

printf("The average of our three numbers is %f\n",avg);

return 0;

}

/\*

avg3 - from Lesson 2a

This program calculates the average of 3 numbers,

Uses an array and a loop, introduces new sum algorithm.

@author Peter N Dobbs

Created: 10/10/2015

\*/

#include <stdio.h>

#include <stdlib.h>

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

// declare variables

float a[3]; //a is a 3 element array

float sum;

float avg;

int i;

// assigning values

a[0] = 3;

a[1] = 5;

a[2] = 8;

//new algorithm

/\*

sum = 0;

i = 0;

while(i<3){

sum = sum + a[i];

//to print int use d (decimal), to print float use f (float)

printf("%d %f\n", i, sum);

i++; // much better than i=i+1

}

\*/

sum = 0;

for(i=0;i<3;i++){

sum = sum + a[i];

}

avg = sum/3;

// print the result

printf("The average of our three numbers is %f\n",avg);

return 0;

}

/\*

avg5 - from Lesson 2b

This program calculates the average of n numbers,

Using an array and a loop, input entered using a text file

The name of the text file has to be in.txt

@author Peter N Dobbs

Created: 10/12/2015

\*/

#include <stdio.h>

#include <stdlib.h>

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

// declare variables

float a[500]; //a is a 3 element array

float sum;

float avg;

int i;

int n;

FILE \*fp;

n = 5; // assign the number of points

//open the file

fp = fopen("in.txt","r"); //r stands for read

if (fp==NULL) { // error catch

printf("File not found\n");

return 1; // ends the program

}

//assigning values

for (i=0;i<n;i++){

fscanf(fp,"%g",&a[i]);

}

//close the file

fclose(fp);

//algorithm for finding sum of array

sum = 0;

for(i=0;i<n;i++){

sum = sum + a[i];

}

avg = sum/(n\*1.0);

// print the result

printf("The average of our %d numbers is %f\n",n,avg);

return 0;

}

/\*

avg6

This program calculates the average of n numbers

in a text file where we don't know how

many numbers are in the file.

This program only works for files < 5000 points

This 'C' program was written by Peter N. Dobbs

10/4/2015

\*/

#include <stdio.h>

#include <stdlib.h>

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

// local variable declaration and initialization

float a[5000]; //array with n elements

float sum = 0; //to be used for sum of array elements

float avg = 0; //to be used for average of array elements

int i = 0; //to be used as index for the for-loops

int num = 0; //to be used for how many numbers are averaged

FILE \*fp; //the star(\*) indicates the pointer.

// open the file

fp = fopen("in.txt", "r"); // 'r': read, 'w': write, 'a': append

if (fp==NULL) { // error catch

printf("File not found\n");

return 0; // ends the program

}

// assigning values to array a[num]

i = 0;

while (fscanf(fp,"%g",&a[i])!=EOF) /\*EOF: End Of File\*/ {

i++;

}

num = i;

// close the file

fclose(fp);

// algorithm for finding sum of array

for (i=0;i<num;i++) {

sum = sum + a[i];

}

// algorithm for finding average of array

avg = sum/num;

// print out results

printf("the average of the %d numbers is %f\n",num,avg);

return 0;

}