//

// main.c

// Chpt4PP2

//

// Created by Randy McMillan on 9/26/13.

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

//

#include <stdio.h>

float weight,height,bmi;

float returnBMI(float w,float h);

void calculate();

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

calculate();

return 0;

}

void calculate(){

printf("Please enter the person's weight in pounds --> ");

scanf("%f",&weight);

printf("Please enter the person's height in inches --> ");

scanf("%f",&height);

bmi = returnBMI(weight, height);

//printf("%f\n",bmi);

if (bmi<18.5) {

printf("The person's BMI = %f and is considered Underweight by the US CDC.\n",bmi);

} else if (bmi < 25) {

printf("The person's BMI = %f and is considered Normal by the US CDC.\n",bmi);

} else if (bmi < 30) {

printf("The person's BMI = %f and is considered Overweight by the US CDC.\n",bmi);

} else {

printf("The person's BMI = %f and is considered Obese by the US CDC.\n",bmi);

}

}

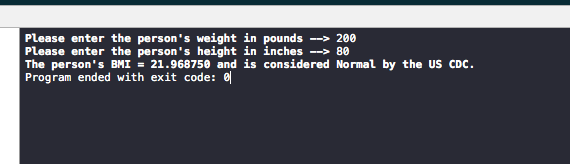
//wt\_lb weight in pounds

//ht\_in height in inches

float returnBMI(float wt\_lb, float ht\_in){

return (703\*wt\_lb)/(ht\_in\*ht\_in);

}



//

// main.c

// Chpt4PP5

//

// Created by Randy McMillan on 9/30/13.

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

//

#include <stdio.h>

#include <string.h>

float n;

void inputRicterScaleNumber();

//

void outputCharacterization(float n);

void outputLittleOrNoDamage(float n);

void outputSomeDamage(float n);

void outputSeriousDamage(float n);

void outputDisaster(float n);

void outputCatastrophy(float n);

void outputSecondPart();

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

inputRicterScaleNumber();

return 0;

}

void inputRicterScaleNumber(){

printf("Please enter the Ricter Scale Number --> ");

scanf("%f",&n);

outputCharacterization(n);

}

void outputCharacterization(float n){

if (n < 5.0) {

outputLittleOrNoDamage(n);

} else if (n >= 5.0 && n < 5.5){

outputSomeDamage(n);

} else if (n >= 5.5 && n < 6.5) {

outputSeriousDamage(n);

} else if (n >= 6.5 && n < 7.5) {

outputDisaster(n);

} else {

outputCatastrophy(n);

}

char yesOrNo[100];

printf("Would you like to calculate another? y/n ");

scanf("%s", yesOrNo);

if (strncmp(yesOrNo, "y", 2) == 0) {

printf("\n");

inputRicterScaleNumber();

}

if (strncmp(yesOrNo, "n", 2) == 0) {

//printf("Good bye!");

char yesOrNo[100];

printf("Would you like to see the answer to the second part of the Assignment? y/n ");

scanf("%s", yesOrNo);

if (strncmp(yesOrNo, "y", 2) == 0) {

printf("\n");

outputSecondPart();

}

if (strncmp(yesOrNo, "n", 2) == 0) {

printf("Good bye!");

}

}

};

void outputLittleOrNoDamage(float n){

printf("Characterization for %.1lf - Little to no damage.\n",n);

}

void outputSomeDamage(float n){

printf("Characterization for %.1lf - Some Damage\n",n);

}

void outputSeriousDamage(float n){

printf("Characterization for %.1lf - Serious Damage: Walls may crack or fail\n",n);

}

void outputDisaster(float n){

printf("Characterization for %.1lf - Disaster: houses and buildings may collapse\n",n);

}

void outputCatastrophy(float n){

printf("Characterization for %.1lf - Catastrophy: most bukldings destroyed\n",n);

}

void outputSecondPart(){

//The second part of the Programming Project answer

printf("In theory a programmer could program a switch to handle all possible Ricter scale numbers. I believe that it is much more prudent to use If/Then statements because it is easier to handle ranges of numbers. By using a statement like:\n\n 'if (n >= 5.0 && n < 5.5){ do someting here }' \n\na wide range of values are handled with much less coding.\n\n\n");

}

