Project Report – Fundamentals of Computing

Tom Penrose

2019 MOD002549 SEM2 F01CAM

2019

# Task 1

This program takes the names of two teams and the scores for each team. It then compares the scores and prints a different message depending on which team scored more or if it was a draw.

#include <stdio.h>

int main() // This function will return an integer

{

char nameTeam1[20]; //Declares variable "nameTeam1" which has 20 characters

char nameTeam2[20]; //Declares variable "nameTeam2" which also has 20 characters

int team1Score, team2Score; //Declares two integer variables "team1Score" and "team2Score"

printf("Please enter the name of team one: "); //Prints a statement for the user, requesting them to enter the desired name of the first team

scanf("%s", nameTeam1); //This line takes the input from the user and assigns it to the variable "nameTeam1"

printf("Please enter the name of team two: "); //Repeats the same process for "nameTeam2" to request and assign the name of the second team

scanf("%s", nameTeam2);

printf("Please enter the score for %s: ", nameTeam1); //Requests the score for the first named team

scanf("%d", & team1Score); //Assigns the user input to variable "team1Score"

printf("Please enter the score for %s: ", nameTeam2); //Repeats the process for "team2Score"

scanf("%d", & team2Score);

if (team1Score > team2Score) //If statement which checks if the integer value for "team1Score" is higher than "team2Score"

{

printf("%s scores 3 points and %s scores 0 points \n", nameTeam1, nameTeam2 ); //If the statement is true print that they score 3 points and the second team scores 0

}

else //If the statement is not true, carrys on to the next if statement

if (team1Score < team2Score) //Compares the two vales to see if the second team's score value is higher than the team's

{

printf("%s scores 3 points and %s scores 0 points \n", nameTeam2, nameTeam1 ); //Print that the second team get 3 points and the first team gets 0

}

else //If neither statements are met then both teams must have the same score

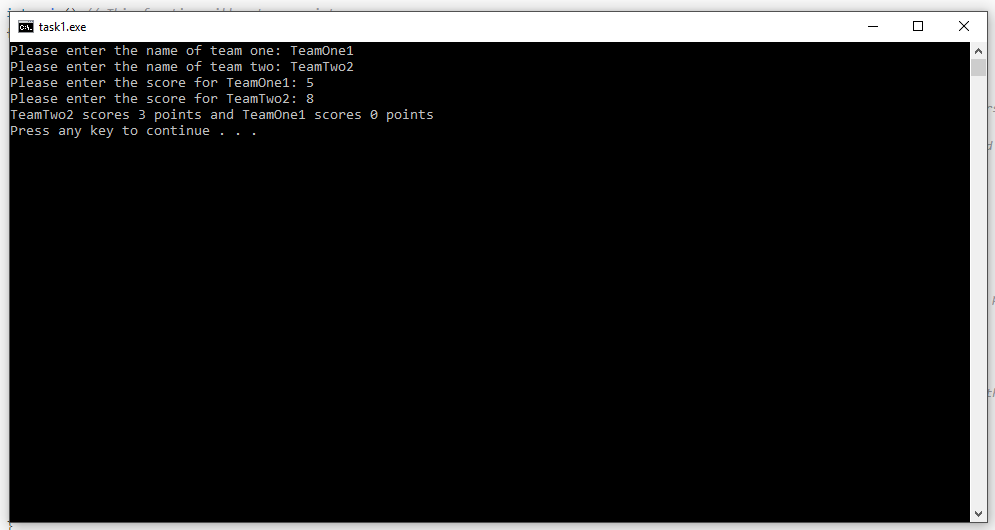
{

printf("Both %s and %s score 1 point \n", nameTeam1, nameTeam2); //Since it must be a draw, both teams recieve 1 point

}

return 0; //The function returns integer 0

}



# Task 2

This program takes student’s names and test scores and assigns a grade based on the mark. The code checks the value of “mark” against the grade boundaries to decide which grade to display. It also increments each counter for grades to keep a list of how many of each grade has been achieved by the students. It then asks if more student information will be inputted and loops.

#include <stdio.h>

int main() //Starts a function that will return an integer

{

char name[20], response, grade; //Initialises 3 string variables

int mark, Fcount=0, Dcount=0, Ccount=0, Bcount=0, Acount=0; //Initialises the marks integer variable

do { //Starts a do while loop

printf("Enter a student name: "); //Asks the user to input the student name

scanf("%s", name); //reads the input from the user and assigns the pointer to the variable "name" as a string

printf("Enter the student's mark: "); //Asks the user to input the student's mark

scanf("%d", &mark); //reads the input from the user and assigns the pointer to the variable "mark" as an integer

printf("%s", name); //Outputs the name of the student from the variable "name"

printf(" recieves the grade: "); //Outputs a string to form a sentence containing the student's name and grade

if (mark < 40){ //Checks if the value of "mark" is less than 40

printf("F"); //If so, the outputted grade is an F

Fcount = Fcount+1;

} //closes the if loop

else if (mark <= 49){ //If the previous statement isn't true, checks to see if "mark" is less than or equal to 49

printf("D"); //If so, the outputted grade is a D

Dcount = Dcount+1;

} //closes the if loop

else if (mark <= 59){ //If the previous statement also isn't true, checks to see if "mark" is less than or equal to 59

printf("C"); //If so, the outputted grade is a C

Ccount = Ccount+1;

} //closes the if loop

else if (mark <= 69){ //If the previous statement also isn't true, checks to see if "mark" is less than or equal to 69

printf("B"); //If so, the outputted grade is a B

Bcount = Bcount+1;

} //closes the if loop

else { //If all the previous statements are false then the grade must be higher than 69 so another if statement is not necessary

printf("A"); //If so, the outputted grade must be an A

Acount = Acount+1;

} //closes the if loop

getchar(); //reads a single character from stdin

printf("\n\nNumber of students with F: %d \nNumber of students with D: %d \nNumber of students with C: %d \nNumber of students with B: %d \nNumber of students with A: %d ", Fcount, Dcount, Ccount, Bcount, Acount);

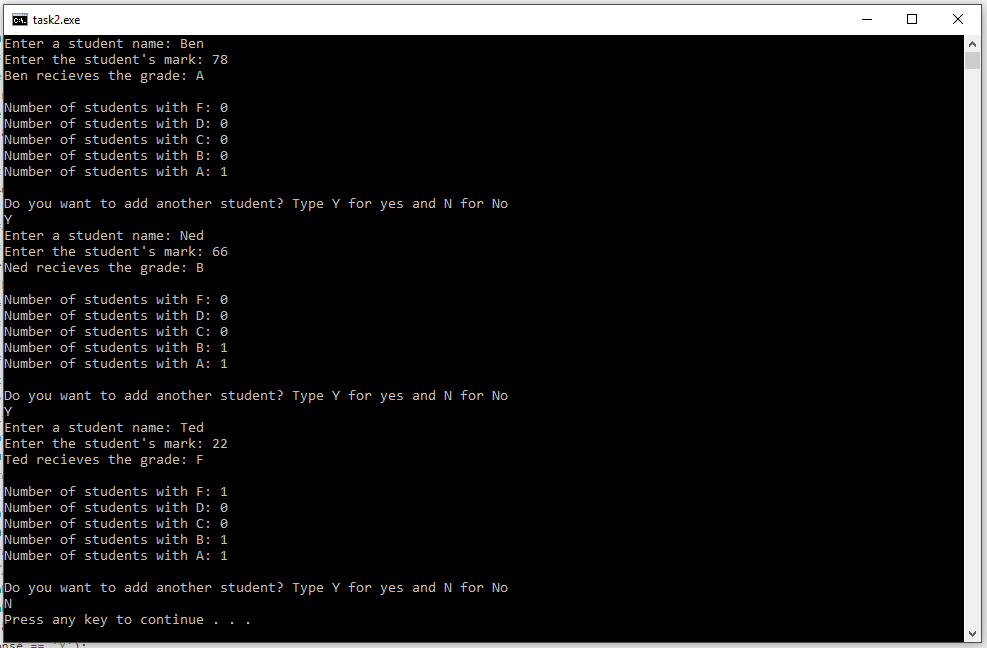
printf("\n\nDo you want to add another student? Type Y for yes and N for No\n");

scanf("%c", &response);

} while (response == 'Y');

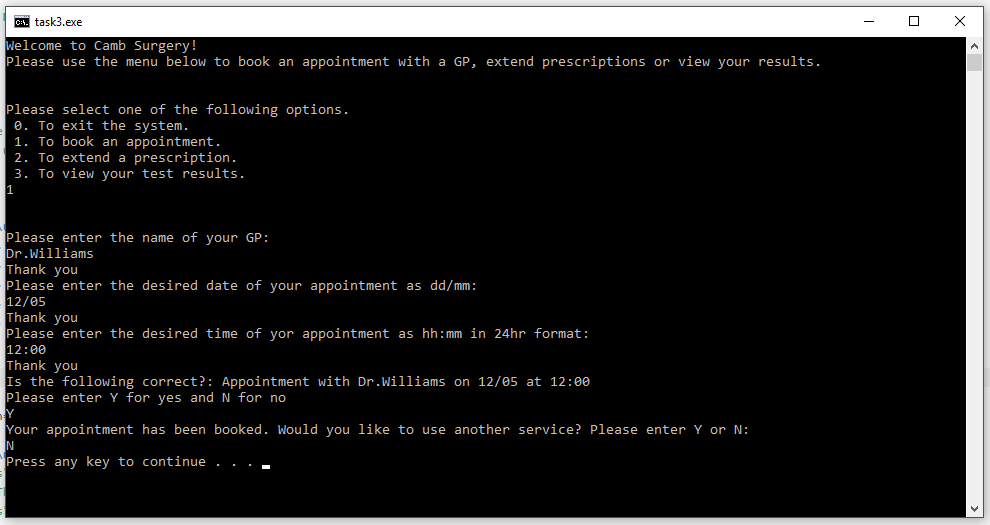
return 0;

}



# Task 3

This program displays a menu for 4 options, exit, booking an appointment, extending a prescription and viewing test results. When selected, the 0th option exits the program with a break command. the 1st option requests the GP’s name, date of appointment and time then repeats it to the user for confirmation. They can then choose to re-input this information or accept it and move on. The program then prompts the user if they’d like to use another service where they can choose to pick from the menu again or close the program with another break. The second option asks for a prescription ID and the length of time they’d like to extend it by in days. It again repeats this information to the user and asks for confirmation before asking to continue with another service. The 3rd option requires the user to input a test ID, asks for confirmation, then “prints” the hypothetical test results and asks if the user would like to use another service again.



#include <stdio.h>

int main()

{

char cont='Y';

char patientName[20];

char dob[10];

char doctorName[20];

char appointmentDate[20];

char time[5];

char response='N';

int option;

char prescID;

int prescETX;

char testID;

printf("Welcome to Camb Surgery!\n");

printf("Please use the menu below to book an appointment with a GP, extend prescriptions or view your results.\n");

//Opening welcome which is only used once so is outside of the for loop

for(;;) { //starts an infinte loop

printf("\n\nPlease select one of the following options.\n");

printf(" 0. To exit the system.\n");

printf(" 1. To book an appointment.\n");

printf(" 2. To extend a prescription.\n");

printf(" 3. To view your test results.\n");

scanf("%d", &option);

//prints out the menu

if (option== 0){

break;

}

else if (option== 1) {

do{ //starts a do while loop

printf("\n\nPlease enter the name of your GP: \n" );

scanf("%s", &doctorName);

printf("Thank you\nPlease enter the desired date of your appointment as dd/mm: \n");

scanf("%s", &appointmentDate);

printf("Thank you\nPlease enter the desired time of yor appointment as hh:mm in 24hr format: \n" );

scanf("%s", &time);

printf("Thank you\nIs the following correct?: Appointment with " );

printf("%s", doctorName);

printf(" on %s", appointmentDate);

printf(" at %s\n", time);

printf("Please enter Y for yes and N for no\n" );

scanf("%c", &response);

}while (response== 'N'); //loops while the response is N

printf("Your appointment has been booked. Would you like to use another service? Please enter Y or N: \n");

scanf("%c", &cont);

if (cont== 'N'){ //exits the for loop if answer is N

break;

}

}

else if (option== 2) {

do{ //starts a do while loop

fflush(stdin); //empties the buffer. To be used before you read a string.

printf("\n\nPlease enter the prescription identifier of the prescription you would like to extend: \n");

scanf(" %c", &prescID);

printf("\nHow many days would you like to extend this prescription for?\n");

scanf("%d", &prescETX);

printf("\nTo confirm, you would like to extend prescription ID: %s for a total of %d days? Please enter Y or yes and N for no\n", prescID, prescETX);

scanf(" %c", &response);

}while (response== 'N'); //loops while the response is N

printf("Your prescription has been extended. Would you like to use another service? Please enter Y or N: \n");

scanf(" %c", &cont);

if (cont== 'N'){ //exits the for loop if answer is N

break;

}

}

else if (option== 3){

do{ //starts a do while loop

printf("\n\nPlease enter your test result identifier: \n");

scanf("%c", &testID);

printf("\nYou have requested the results for test: %s. Is this correct? Please enter Y or yes and N for no\n", testID);

scanf("%c", &response);

}while (response== 'N'); //loops while the response is N

printf("\n The results of your test are now printing. Would you like to use another service? Please enter Y or N: \n");

scanf("%c", &cont);

if (cont== 'N'){ //exits the for loop if answer is N

break;

}

}

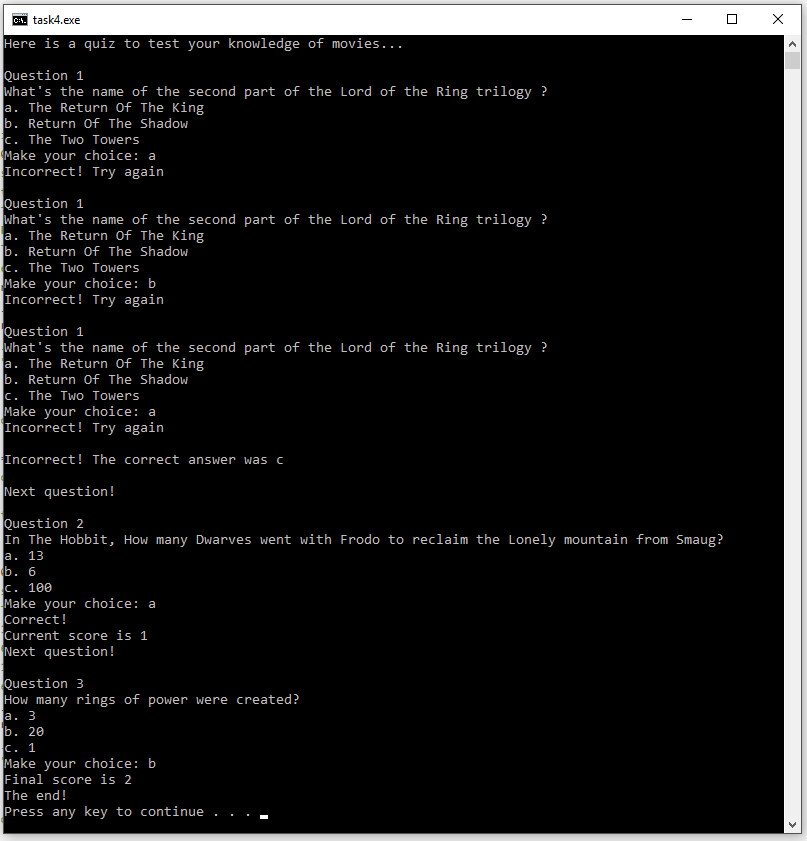
}

return 0; //ends the program

}

# Task 4

This program is a quiz that gives the user 3 question to answer, picking from 3 possible answers and giving 3 chances to get it correct before moving on. The variable “chances” is set before each question and iterates every time the question is answered. If it reaches 3, the correct answer is revealed and the question is skipped. If the correct answer is entered, the “score” variable is incremented and the next question is displayed. After all 3 questions, the end score is revealed.



#include <stdio.h>

int main()

{

char answer;

int chances;

int score=0;

printf("Here is a quiz to test your knowledge of movies...\n\n");

for (chances=0; chances < 3; ++chances){ //starts a for loop which only lets the user try 3 times

printf("Question 1\n");

printf("What's the name of the second part of the Lord of the Ring trilogy ?\n");

printf("a. The Return Of The King\n");

printf("b. Return Of The Shadow\n");

printf("c. The Two Towers\n");

printf("Make your choice: ");

fflush(stdin);

scanf(" %c", &answer);

if (answer == 'c'){ //increments the score and breaks the for loop when recieving correct answer

++score;

printf("Correct!\nCurrent score is %d \n", score);

break;

}

printf("Incorrect! Try again\n\n");

}

if (chances == 3){ //gives the user the answer if they failed the 3 chances

printf("Incorrect! The correct answer was c \n");

}

printf("\nNext question!\n\n");

for (chances=0; chances < 3; ++chances){ //starts a for loop which only lets the user try 3 times

printf("Question 2\n");

printf("In The Hobbit, How many Dwarves went with Frodo to reclaim the Lonely mountain from Smaug?\n");

printf("a. 13\n");

printf("b. 6\n");

printf("c. 100\n");

printf("Make your choice: ");

scanf(" %c", &answer);

if (answer == 'a'){ //increments the score and breaks the for loop when recieving correct answer

++score;

printf("Correct!\nCurrent score is %d", score);

break;

}

printf("Incorrect! Try again\n\n");

}

if (chances == 3){ //gives the user the answer if they failed the 3 chances

printf("Incorrect! The correct answer was a \n");

}

printf("\nNext question!\n\n");

for (chances=0; chances < 3; ++chances){ //starts a for loop which only lets the user try 3 times

printf("Question 3\n");

printf("How many rings of power were created?\n");

printf("a. 3\n");

printf("b. 20\n");

printf("c. 1\n");

printf("Make your choice: ");

scanf(" %c", &answer); //increments the score and breaks the for loop when recieving correct answer

if (answer == 'b'){

++score;

break;

}

printf("Incorrect! Try again\n\n");

}

if (chances == 3){ //gives the user the answer if they failed the 3 chances

printf("Incorrect! The correct answer was b \n");

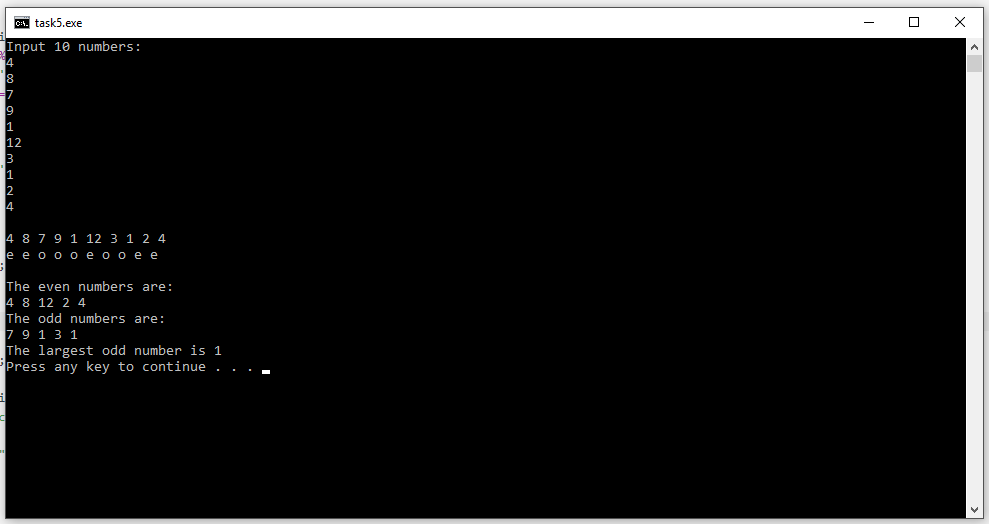
}

printf("Final score is %d", score); //prints the total score and ends the program

printf("\nThe end!\n");

}

# Task 5

This program finds the largest odd number out of 10 inputted numbers. First, it checks each number in the array to see if it’s odd or even. It prints the numbers and an ‘o’ or ‘e’ to show if they are odd or even. It then sorts them into two separate arrays, one for even numbers and one for odd numbers and displays both arrays. Lastly, it compares every value in the odd array to find the largest value and assigns that to the ‘big’ value which is then printed. If there were no odd numbers, then the value of ‘big’ will be the same as what it was when it was initialised, zero, so it prints that there were no odd numbers.

#include <stdio.h>

int main()

{

int arr[10], i, j, x=0, y=0, even[10], odd[10], big=0;

char oe[10];

printf("Input 10 numbers:\n");

for (i = 0; i < 10; i++){ //loops to input 10 numbers into the array

scanf(" %d", &arr[i]);

}

for (i = 0; i < 10; i++){ //checks if each value in the array is divisible by 2, therefore even, and puts it into the even array

if(arr[i] % 2 == 0){

oe[i] = 'e';

even[y] = arr[i];

y++;

}

else{ //every other value goes into the odd array

oe[i] = 'o';

odd[x] = arr[i];

x++;

}

}

printf("\n");

for (i = 0; i < 10; i++){ //displays all the numbers

printf("%d ", arr[i]);

}

printf("\n");

for (i = 0; i < 10; i++){ //displays wether each is odd or even with 'o' or 'e'

printf("%c ", oe[i]);

}

printf("\n\n");

printf("The even numbers are: \n"); //prints all the numbers from even array

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

printf("%d ", even[i]);

}

printf("\n");

printf("The odd numbers are: \n"); //prints all the numbers from odd array

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

printf("%d ", odd[i]);

}

printf("\n");

for (i = 0; i < x; i++){ //compares each odd number to find the biggest

if(odd[i]>odd[i+1])

big = odd[i];

}

if (big==0){ //if big = 0 then no odd numbers

printf("There were no odd numbers");

}else{

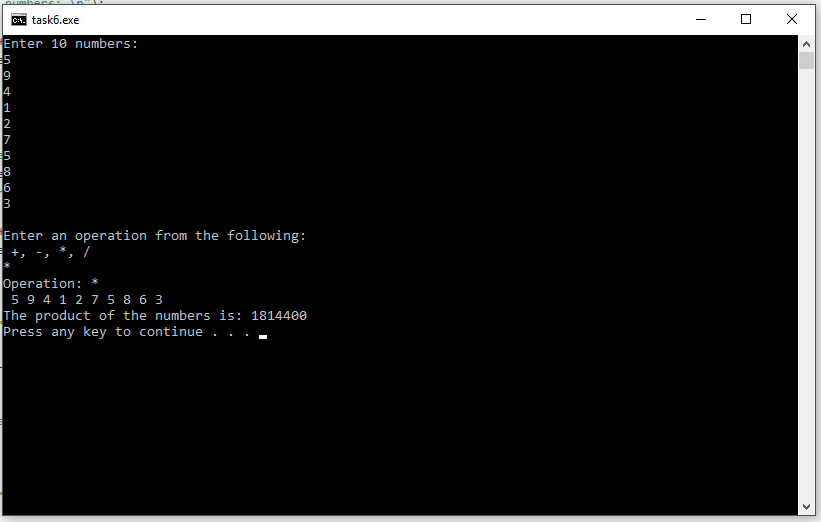
printf("The largest odd number is %d", big); //prints biggest odd number

}

return 0;

}

# Task 6

This program takes 10 values then performs an operation on them chosen by the user. First, the 10 values are inputted by the user and stored in an array. The user then enters one of the 4 operators shown in the console and the operation is completed on the values in the array. Depending on the operation, each number from the array is added, subtracted, multiplied or divided from each other to produce the end value which is then printed.

#include <stdio.h>

int main()

{

int arr[10], x, sum=0, mult=0, div=0, sub=0;

char operation;

printf("Enter 10 numbers: \n");

for(x=0; x<10; x++){ //loops to input 10 numbers into the array

scanf(" %d", &arr[x]);

}

do{ //starts a do while loop

printf("\nEnter an operation from the following: \n +, -, \*, / \n");

scanf(" %c", &operation);

printf("Operation: %c \n", operation); //user inputs operation into 'operation' char

for(x=0; x<10; x++){ //prints the array

printf(" %d", arr[x]);

}

if(operation == '+'){ //if add is chosen, sum of all the numbers in array

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

sum=arr[x]+sum;

}

printf("\nThe sum of the numbers is: %d", sum); //prints answer and break

break;

}

if(operation == '-'){ //subtracts each number in the array

sub=arr[0];

for(x=1; x<10; x++){

sub=sub-arr[x];

}

printf("\nThe numbers subtracted from eachother equal to: %d", sub); //prints answer and breaks

break;

}

if(operation == '\*'){ //multiplies each number in the array

mult=arr[0];

for(x=1; x<10; x++){

mult=arr[x]\*mult;

}

printf("\nThe product of the numbers is: %d", mult); //prints answer and breaks

break;

}

if(operation =='/'){ //divides each number in the array

div=arr[0];

for(x=1; x<10; x++){

div=div/arr[x];

}

printf("\nEach number divided by the next is equal to: %d", div); //prints answer and breaks

break;

}

}while(operation != ('+'|'-'|'\*'|'/')); //if anything else is entered, loop exits

return 0;

}

# Task 7

This program prints the smallest possible number from a set of inputted numbers. First, the number of values is inputted so we know how many times to iterate in a later section. Next, the numbers are entered and go into an array. Each value in this array is now checked to see if it’s positive. If so, it goes into a separate array for positive numbers. This array is then fed into the “smallest” function that checks one value against another and returns the smallest. This is repeated for the whole positive array then the smallest positive number is printed.

#include <stdio.h>

int smallest(int num1, int num2)

{

if(num1 > num2){

return num2;

}else

return num1;

} //function takes two integers, compares the two and returns the smallest value

int main()

{

int arr[10], pos[10], i, x, j=0, small;

printf("How many numbers?:");

scanf(" %d", &x); //number of times to loop the for

printf("\nInput %d numbers:\n", x); //recieves and stores numbers in an array

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

scanf(" %d", &arr[i]);

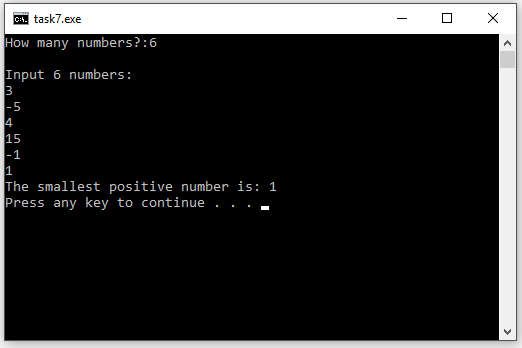
}

for (i = 0; i < x; i++){ //takes the array and moves positive values into another array

if(arr[i]>0){

pos[j] = arr[i];

j++;

 }

}

small = pos[0];

for (i = 1; i < j; i++){

small = smallest(small, pos[i]); //calls the function with each value of positive array and saves to 'small'

}

printf("The smallest positive number is: %d", small); //prints the 'small' int value

return 0;

}

# Task 8

This program serves as a system to store, process and display student’s marks. First, the main function starts an infinite loop in which it calls the “printMenu” function to display the menu and options. The 1st option runs the function “students” which asks for the user to input the number of students in the class and their names and marks respectively by putting them into two arrays, “name” and “mark”. The 2nd option calls the function “printStudents” which prints out the student’s names and marks from these arrays. The 3rd option calls the “highMark” function which checks each value in the mark array and finds the largest number. It then takes that number and copies the location of the largest value in the mark array and uses it to find the name of that student. It then prints both the name and mark of the student. The 4th option calls the function “highMark” which takes all the student’s marks and creates an average for the class by adding each number together then dividing by the total number of students. It then prints this value. The 5th and final option exits the program with a break.

#include <stdio.h>

#include <string.h>

int x, i, option, stID, highest, avg=0;

int mark[30];

char name[30][10];

void printMenu()

{ //function that prits the menu

printf("\nPlease select one of the following options.\n");

printf(" 1. Input students' names and marks\n");

printf(" 2. Display all the students' names and marks\n");

printf(" 3. Display the highest mark in the class\n");

printf(" 4. Display average class mark\n");

printf(" 5. Exit\n");

}

void students()

{ //function that takes the student's names and marks and puts them into arrays

printf("How many students are in the class? (max 30): ");

scanf("%d", &x);

printf("\nInput %d student's names:\n", x);

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

fflush(stdin);

scanf("%s", name[i]);

}

printf("\nPlease enter the marks for each student\n");

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

printf("%s: ", name[i]);

scanf(" %d", &mark[i]);

printf("\n");

}

}

void printStudents()

{ //function that prints all the info from student arrays

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

printf("%s: ", name[i]);

printf("%d\n", mark[i]);

}

}

void highMark()

{ //function that finds the highest mark and student name

highest = mark[0];

for (i = 1; i < x; i++){ //loops for all students

if (mark[i] > highest){ //iterates over every mark

highest = mark[i];

stID = i; //copies array ID to find the correct student name

}

}

printf("The student with the highest mark is %s with a mark of: %d\n", name[stID], highest); //prints mark and name

}

void averageMark()

{ //fucniton that averages all the marks and prints the values

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

avg += mark[i]; //loops adding each value together

}

avg /= x; //divides from amount of values in array

printf("The average mark of the class is: %d\n", avg); //prints average

}

int main ()

{

for(;;){ //starts an infinte loop

printMenu(); //calls the menu function then recieves option input

scanf("%d", &option);

if (option==1){

students(); //calls function

}else if (option==2){

printStudents(); //calls function

}else if (option==3){

highMark(); //calls function

}else if (option==4){

averageMark(); //calls function

}else if (option==5){

break; //breaks from infinite loop

}

}

return 0;

}

