Table of Contents

**SIMPLE CALCULATION1**

WRITTEN CODE2

CODE3

RESULT……………………………………………………………………………………………………………………………………………4

**AGE CALCULATION5**

CODE6

RESULT7

Simple calculator

Is a program that can perform arithmetic operations for example addition, subtraction, multiplication, and division. This program user input for

Give output that can be given by calculator.

**User interaction**: The program users want to enter an arithmetic operator and two numerical operands to give the output.

**Arithmetic operations**: This are the syntax used in C language when we write the code.

**Addition (+)**: While two numbers addition.

**Subtraction (-)**: Used for difference between two numbers while subtraction .

**Multiplication (\*)**: Used for multiplies two numbers.

**Division (/)**: Divides one number by another when error occur for division by zero.

**Control Structures**: It utilizes a switch statement to determine which operation to perform based on the user's input. This helps demonstrate how to manage different conditions in programming.

**Error while running program**: The program includes basic error handling to manage invalid operator inputs and prevent division by zero, enhancing user experience and robustness.

CODE OF SIMPLE CALCULATION

Written code

#include <stdio.h>

Int main () {

*char* op; // this is character name.

*double* num1, num2;

    printf("Enter an operator (+, -, \*, /): ");

    scanf("%c", &op);

    printf("Enter two operands: ");

    scanf("%lf %lf", &num1, &num2);

    switch(op) {

        case '+':

            printf("%.1lf + %.1lf = %.1lf\n", num1, num2, num1 + num2);

            break;

        case '-':

            printf("%.1lf - %.1lf = %.1lf\n",num1, num2, num1 - num2);

            break;

        case '\*':

            printf("%.1lf \* %.1lf = %.1lf\n", num1, num2, num1 \* num2);

            break;

case '/':

            if (num2 != 0)

                printf("%.1lf / %.1lf = %.1lf\n", num1, num2, num1/ num2);

            else

                printf("wrong division by zero.\n");

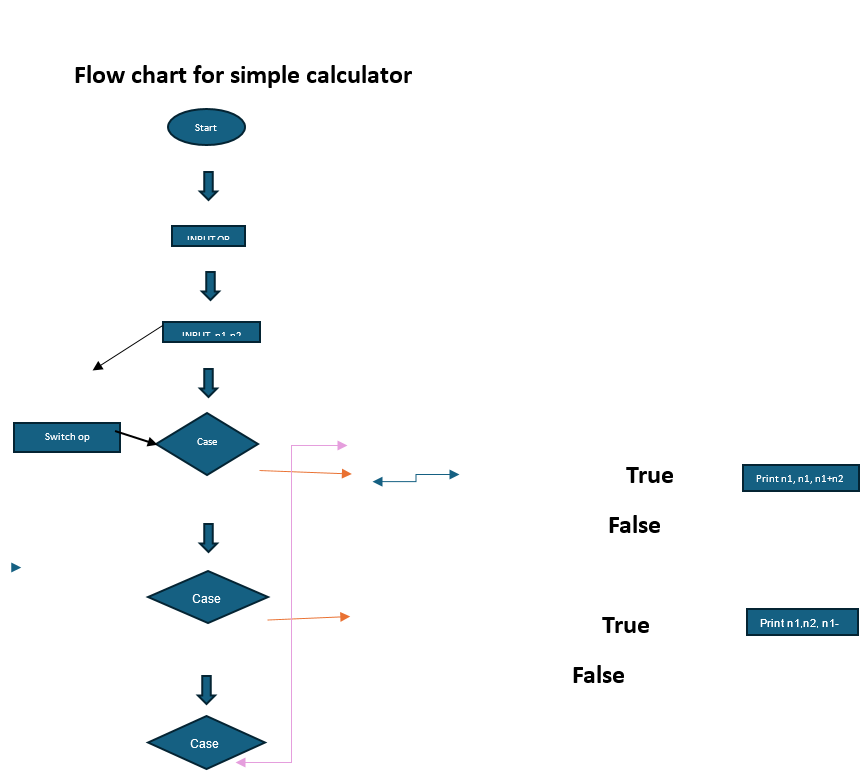
            break;

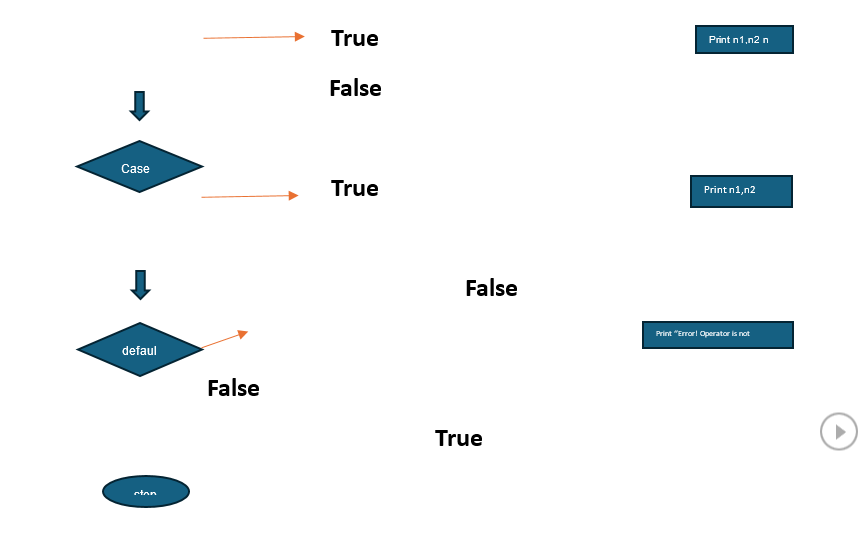
        default:

    printf("wrong! operator is not correct.\n");

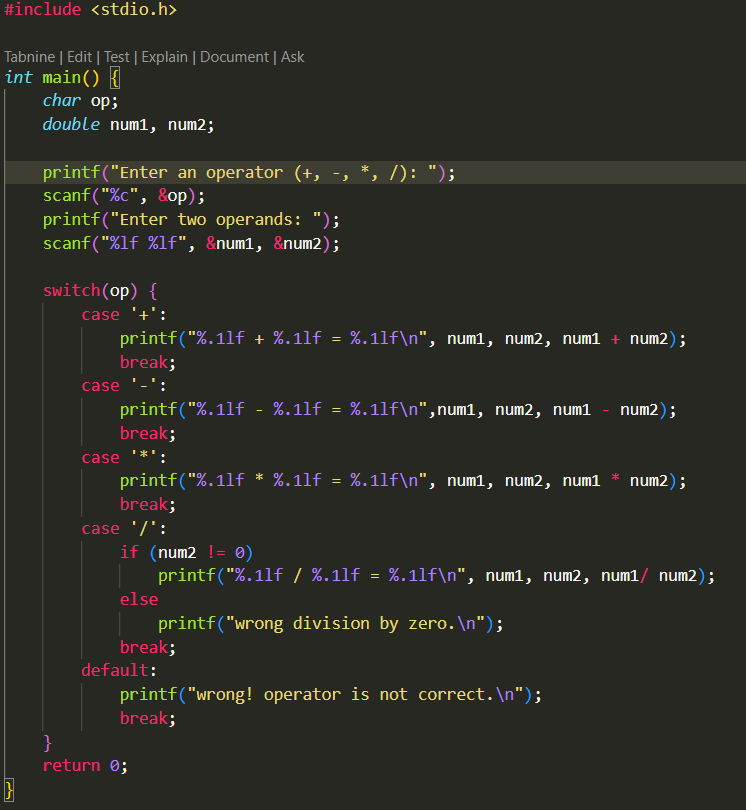
break;    }

 return 0 }

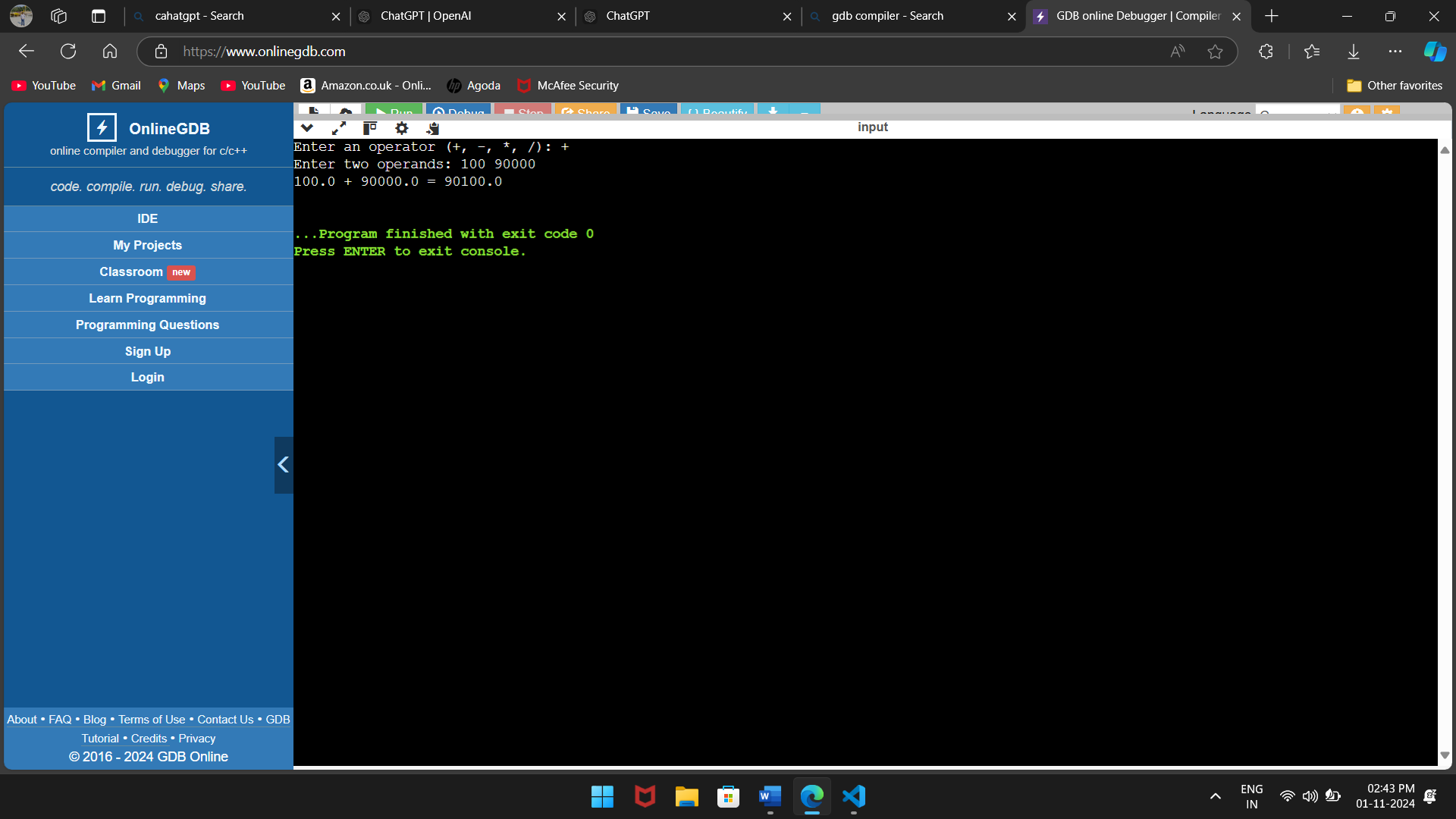




CODE



RESULT



**How It Works:**

1. **Input**: The program prompts the user to input an operator and two numerical values.
2. **Processing**: Using a switch statement, it determines which arithmetic operation to perform based on the input operator.
3. **Output**: It calculates the result and prints it to the console, formatted to one decimal place. It also handles cases for invalid operators and division by zero.

**Compilation**

To compile and run this program:

1. Save the code in a file named calculator.c.
2. Open a terminal and navigate to the directory where the file is saved.
3. Compile the code using gcc calculator.c -o calculator.
4. Run the program with .calculat

AGE CALCULATION

INTRODUCTION

An age calculator program in an application that helps users can declare their age on their date of birth and the current date. This program can be including user input date handling arithmetic calculate in this program.

* 1. **User Input**: The program contains users’ data to enter their birth date typically including the day month and year. It can also obtain the current date from the system or request it from the user.
  2. **Date Calculation**: It calculates the age by comparing the provided birth date with the current date. This involves handling various scenarios such as leap years and the differing number of days in each month.
  3. **Detailed Output**: The output usually displays the age in years months and days offering a clear and informative result.
  4. **Error Handling**: A well-designed program may include validation to ensure that the entered dates are realistic preventing invalid inputs like future dates or incorrect format

WRITTEN CODE

**#include <stdio.h>**

***int* main () {**

***int* d1,m1,y1,d2,m2,y2,r1,r2,r3;**

**//input the birthday and current date**

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

**printf("                         Age calculator                         \n");**

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

**printf("enter your birthday day");**

**scanf("%d",&d1);**

**printf("enter your birthday month");**

**scanf("%d",&m1);**

**printf("enter your birthday year");**

**scanf("%d",&y1);**

**printf("enter the current day");**

**scanf("%d",&d2);**

**printf("enter the current month");**

**scanf("%d",&m2);**

**printf("enter the current year");**

**scanf("%d",&y2);**

**//calculate the age**

**r1 = y2 - y1;**

**r2 = m2 - m1;**

**r3 = d2 - d1;**

**if(r3<0)**

**{**

**r2--;**

**r3 += 30;**

**}**

**if(r2<0){**

**r1--;**

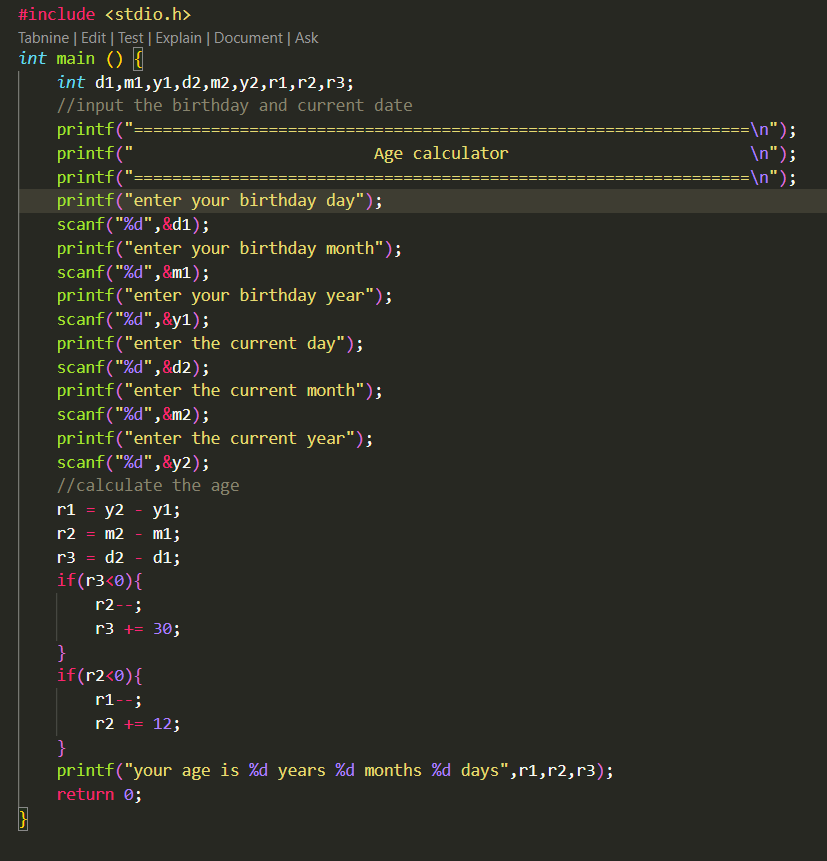
**r2 += 12;**

**}**

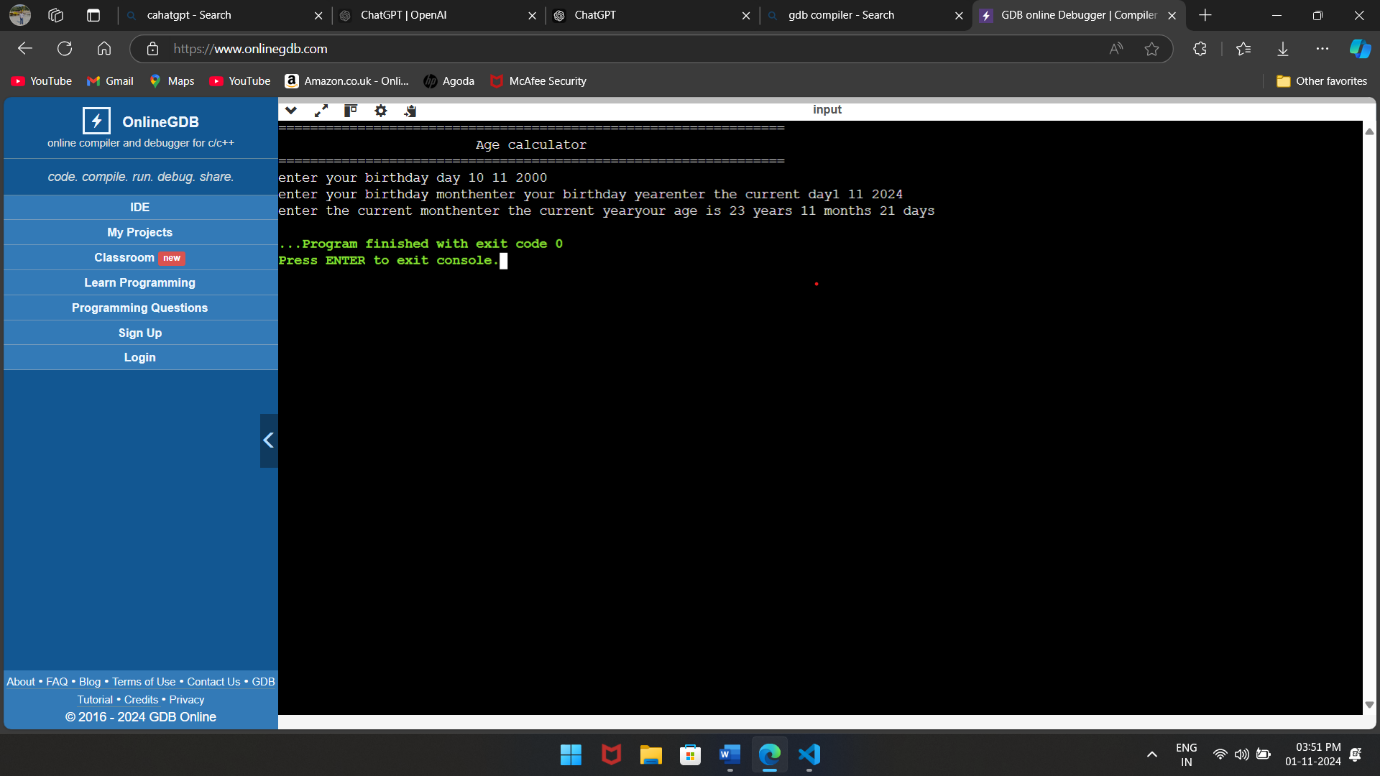
**printf("your age is %d years %d months %d days",r1,r2,r3);**

**return 0;**

**}**

**CODE** 

**RESULT**



**How It Works:**

**1 Enter day of birthday.**

**2 Enter month of birthday.**

**3 Enter year of birthday.**

**4 Enter current day.**

**5Enter current month.**

**6Enter current year.**

**This is the working of age calculator.**

**Purpose:**

* Input and output functions
* Conditional statements and loops
* Arithmetic operations
* Date and time manipulation