DAY 8-DAILY ASSIGNMENTS

ANSU MARIUM SHIBU

15-11-2024

1. Create a C program that defines a function to increment an integer by 1. The function should demonstrate call by value, showing that the original value remains unchanged.

```c
#include<stdio.h>

void increment_one(int);

int main(){
    int num=2;
    printf("original val:%d\n",num);

    increment_one(num);

    printf("value after call:%'d\n ",num);
    return 0;
}

void increment_one(int a){
    a=a+1;
    printf("value after increments:%d\n",a);
}
```

```
PS D:\c progrms coding> gcc funass1.c
PS D:\c progrms coding> ./a
original val:2
value after increments:3
value after call:2
```

2. Write a C program that attempts to swap two integers using a function that employs call by value. Show that the original values remain unchanged after the function call.

```c
#include<stdio.h>

void swap(int,int);

int main(){
    int a=5,b=1;
    printf("original val:a=%d\n b=%d\n",a,b);

    swap(a,b);

    printf("value swap after call:a=%'d\n b=%d\n ",a,b);
    return 0;
}

void swap(int x,int y){
    int temp;
    temp=x;
    x=y;
    y=temp;
    printf("value inside fun:x=%d\n y=%d\n",x,y);
}
```

```
PS D:\c progrms coding> gcc funassi2.c
PS D:\c progrms coding> ./a
original val:a=5
 b=1
value inside fun:x=1
 y=5
value swap after call:a=5
 b=1
```

3. Develop a C program that calculates the factorial of a number using call by value.

```c
#include<stdio.h>

int factorial(int);

int main(){
    int num;
    printf("enter num:\n");
    scanf("%d",&num);

    int res=factorial(num);

    printf("factorial %d:%d\n",num,res);
    return 0;
}
int factorial(int n){
    if (n==1)
        return 1;

    else

        return n*factorial(n-1);
```

```
PS D:\c progrms coding> gcc funassi3.
PS D:\c progrms coding> ./a
enter num:
3
factorial 3:6
3
factorial 3:6
```

4. Create a C program that defines a function to find the maximum of two numbers using call by value.

```c
#include<stdio.h>

int max(int, int);

int main() {
    int num1, num2;
    printf("Enter two numbers: ");
    scanf("%d %d", &num1, &num2);
    int result = max(num1, num2);
    printf("The maximum of %d and %d is: %d\n", num1, num2, result);
    return 0;
}

int max(int a, int b) {
    if(a > b)
        return a;
    else
        return b;
}
```

```
PS D:\c progrms coding> gcc funassi4.c
PS D:\c progrms coding> ./a
PS D:\c progrms coding> ./a
Enter two numbers: 4
1
The maximum of 4 and 1 is: 4
```

5.Problem Statement 1: Arithmetic Operations Calculator

Description: Write a C program that performs basic arithmetic operations (addition, subtraction, multiplication, and division) on two numbers provided by the user. The program should use functions to perform each operation and demonstrate call by value.

Requirements:

Create separate functions for addition, subtraction, multiplication, and division.

Each function should take two parameters (the numbers) and return the result.

Use appropriate data types for the variables.

Use operators for arithmetic calculations.

Example Input/Output:

Enter first number: 10

Enter second number: 5

Addition: 15

Subtraction: 5

Multiplication: 50

Division: 2.0

```c
#include<stdio.h>


int add(int, int);
int sub(int,int);
int mul(int,int);
float div(int,int);

int main(){

    int num1,num2;
    printf("entr frst nums:\n");
    scanf("%d",&num1);
    printf("enter sec num:\n");
    scanf("%d",&num2);

    printf("add:%d\n",add(num1,num2));
    printf("sub:%d\n",sub(num1,num2));
    printf("mul:%d\n",mul(num1,num2));


    if(num2!=0){
      printf("div:%.2f\n",div(num1,num2));
    }else{
      printf("div:invalid\n");
    }
    return 0;
```

```
28
29
30  }
31  int add(int a,int b){
32      return a+b;
33  }
34  int sub(int a,int b){
35      return a-b;
36  }
37  int mul(int a,int b){
38      return a*b;
39  }
40
41  float div(int a,int b){
42      return (float)a/b;
43  }
```

```
PS D:\c progrms coding> gcc funtas1.c
PS D:\c progrms coding> ./a
entr frst nums:
4
enter sec num:
6
add:10
sub:-2
mul:24
div:0.67
```

6. Problem Statement 2: Temperature Conversion

Description: Develop a C program that converts temperatures between Celsius and Fahrenheit. The program should use functions to handle the conversions and demonstrate call by value.

Requirements:

Create two functions: one for converting Celsius to Fahrenheit and another for converting Fahrenheit to Celsius.

Each function should accept a temperature value as an argument and return the converted temperature.

Use appropriate data types for temperature values.

Use arithmetic operators to perform the conversion calculations.

Example Input/Output:

Enter temperature in Celsius: 25

Temperature in Fahrenheit: 77.0

Enter temperature in Fahrenheit: 77

Temperature in Celsius: 25.0

```c
#include<stdio.h>

float cel_fah(float);
float fah_cel(float);

int main(){
    float temp;
    int choice;


    printf("Temperature Conversion Program\n");
    printf("1.celsius to fahrenheit\n");
    printf("2.fahrenheit to celsius\n");
    printf("enter the choice:\n");
    scanf("%d",&choice);

    if(choice==1)
    {
        printf("enter temp in celsius:");
        scanf("%f",&temp);
        printf("temp in Fahrenheit:%.2f\n",cel_fah(temp));
    }
    else if(choice==2){
        printf("entr temp in Fahrenheit:");
        scanf("%f",&temp);
        printf("temp in celsius:%.2f\n",fah_cel(temp));
    }
    else{
        printf("invalid\n");
```

```c
        printf("entr temp in Fahrenheit:");
        scanf("%f",&temp);
        printf("temp in celsius:%.2f\n",fah_cel(temp));
    }
    else{
        printf("invalid\n");
    }
    return 0;
}

float cel_fah(float celsius){
    return (celsius * 9 / 5) + 32;
}
float fah_cel(float fahrenheit){
    return (fahrenheit - 32) * 5 / 9;
}
```

```
PS D:\c progrms coding> gcc funntas2.
PS D:\c progrms coding> ./a
Temperature Conversion Program
1.celsius to fahrenheit
2.fahrenheit to celsius
enter the choice:
1
enter temp in celsius:25
temp in Fahrenheit:77.00
```

7. Problem Statement 2: Simple Interest Calculator

Description: Develop a C program that calculates simple interest based on user input for principal amount, rate of interest, and time period. The program should use a function to compute interest and demonstrate call by value.

Requirements:

Implement a function that takes three parameters (principal, rate, time) and returns the calculated simple interest.

Use appropriate data types for financial calculations (e.g., float or double).

Utilize arithmetic operators to compute simple interest using the formula

SI = P×R×T/100

Example Input/Output:

Enter principal amount: 1000

Enter rate of interest: 5

Enter time period (in years): 3

Simple Interest is: 150.0

```c
#include<stdio.h>

float simple_int(float pr, float rate,float time);

int main(){
    float pr,rate,time,interest;

    printf("enter pr amount:");
    scanf("%f",&pr);

    printf("enter rate:");
    scanf("%f",&rate);

    printf("eneter time:");
    scanf("%f",&time);

    interest=simple_int(pr,rate,time);

    printf("simple interest:%.2f\n",interest);

    return 0;
}

float simple_int(float pr, float rate,float time)
{
```

```c
    return 0;
}

float simple_int(float pr, float rate,float time)
{
    return (pr*rate*time)/100;
}
```

```
PS D:\c progrms coding> gcc funtas3.c
PS D:\c progrms coding> ./a
enter pr amount:230
enter rate:5
enter pr amount:230
enter rate:5
enter rate:5
eneter time:10
simple interest:115.00
PS D:\c progrms coding>
```

8. 1) Create a char type variable and initialize it to value 100

2) Print the address of the above variable.

3) Create a pointer variable and store the address of the above variable

4) Perform read operation on the pointer variable to fetch 1 byte of data from the pointer

5) Print the data obtained from the read operation on the pointer.

6) Perform write operation on the pointer to store the value 65

7) Print the value of the variable defined in step 1

```c
#include<stdio.h>

int main(){
    char ch=100;
    printf("address of var:%p\n",&ch);

    char *ptr=&ch;
    char data=*ptr;
    printf("data from pointer:%d\n",data);
    *ptr=65;
    printf("update val ch:%d\n",ch);


}
```

```
PS D:\c progrms coding> gcc pointertas.c
PS D:\c progrms coding> ./a
address of var:00000078eefff8e6
data from pointer:100
update val ch:65
```

9. Write a C program that swaps the values of two integers using pointers.

```c
#include<stdio.h>

int main(){
    int a,b,*p1,*p2,temp;

    printf("enter frst num:");
    scanf("%d",&a);
    printf("enetr sec num:");
    scanf("%d",&b);

    p1=&a;
    p2=&b;
    printf("before swap: a=%d, b=%d\n",*p1,*p2);
    temp=*p1;
    *p1=*p2;
    *p2=temp;

    printf("after swap: a=%d,b=%d\n",*p1,*p2);

}
```

```
PS D:\c progrms coding> gcc point
PS D:\c progrms coding> ./a
enter frst num: 5
enetr sec num:4
enetr sec num:4
before swap: a=5, b=4
after swap: a=4,b=5
PS D:\c progrms coding>
```

10. WAP to swap the number using swap function and follow the pass by reference method.

```c
#include<stdio.h>

void swap(int *x,int *y);

int main(){
    int a=10,b=20;

    printf("before swap:\n");
    printf("a=%d , b=%d\n",a,b);

    swap(&a,&b);

    printf("after swap:\n");
   printf("a=%d , b=%d\n",a,b);

}
void swap(int *x,int *y){
    int temp;
    temp=*x;
    *x=*y;
    *y=temp;
}
```

```
PS D:\c progrms coding> gcc callref
PS D:\c progrms coding> ./a
before swap:
a=10 , b=20
after swap:
a=20 , b=10
PS D:\c progrms coding> gcc callref
```

11. WAP for Finding the Cube of a Number Using Pass by Reference

```c
#include<stdio.h>

void cube(int *n);
int main(){
    int num;
    printf("enter num:");
    scanf("%d",&num);

    cube(&num);
    printf("vube of num:%d\n",num);
    return 0;
}

void cube(int *n){
    *n = (*n) * (*n) * (*n);
}
```

```
PS D:\c progrms coding> gcc callreftas2.c
PS D:\c progrms coding> ./a
enter num:4
vube of num:64
```

12. WAP to calculate the simple interest with the help of a function and pass call by reference method.

```c
#include<stdio.h>

void simple_int(float *pr, float *rate, float *time, float *interest);

int main() {
    float pr, rate, time, interest;

    printf("Enter principal amount: ");
    scanf("%f", &pr);

    printf("Enter rate: ");
    scanf("%f", &rate);

    printf("Enter time: ");
    scanf("%f", &time);

    simple_int(&pr, &rate, &time, &interest);

    printf("Simple Interest: %.2f\n", interest);

    return 0;
```

```c
}

void simple_int(float *pr, float *rate, float *time, float *interest) {
    *interest = (*pr) * (*rate) * (*time) / 100;
}
```

```
PS D:\c progrms coding> gcc callreftas3.c
PS D:\c progrms coding> ./a
Enter principal amount: 400
Enter rate: 4
Enter time: 3
Enter time: 3
Simple Interest: 48.00
PS D:\c progrms coding>
```