# Phase 1 Day 3(8/11/24)

## 1. Variable Initialization

Question: Write a program that declares an integer variable, initializes it with a value of 42, and prints it to the console.

#include<stdio.h>

int main()

{

int  a=42;

printf("a= %d",a);

}

## 2. Swapping Variables

Question: Create a program that swaps the values of two integer variables without using a temporary variable. Demonstrate this by printinthg e values before and after the swap.

#include<stdio.h>

int main()

{

   int num1,num2;

   printf("Enter the two numbers: ");

   scanf("%d %d",&num1,&num2);

   printf("Before swapping\nnum1=%d\nnum2=%d",num1,num2);

   num1=num1+num2;//num1=2 num2=4 here num1=6

   num2=num1-num2;//6-4=2

   num1=num1-num2;//6-2=4

   printf("\nAfter swapping \nnum1=%d\nnum2=%d",num1,num2);


}

# 3. User Input and Output

Question: Write a program that prompts the user to enter their name and age, stores these values in appropriate variables, and then prints a greeting message that includes both the name and age.

```c
#include<stdio.h>

int main()

{

    char name[10];

    int age;

printf("Hi,User!\n Please enter your Name: ");

scanf("%s",&name);

printf("Enter your age: ");

scanf("%d",&age);

printf("Name of user: %s \nAge:%d\n",name,age);

}
```

# 4. Data Type Conversion

Question: Write a program that declares an integer variable, assigns it a value of 10, and then converts it to a float variable. Print both the integer and float values to show the conversion

```c
#include<stdio.h>

int main()

{

    int a=10;

    float b=(float) a;

    printf("integer value :%d\n",a);
```

```
    printf("Float value :%f",b);

    return 0;

}
```
.

# 5. Constants vs. Variables

Question: Using `#define`, create a constant for the value of Pi (3.14). Write a program that calculates the area of a circle given its radius (stored in a variable) and prints the result using the constant for Pi.

```
#include<stdio.h>

#define Pi 3.14

int main()

{

    int r;

    float area;

    printf("Enter the radius of the circle : ");

    scanf("%d",&r);

    area=Pi*r*r;

    printf("Area=%g",area);

}
```

# 6. Scope of Variables

Question: Write a program that demonstrates the concept of variable scope by declaring a global variable and modifying it within a function. Print the value of the global variable before and after modification.

```
#include<stdio.h>

int a=10; //global variable

void modify()

{
```

```c
    a+=200;

    printf("\nModified value of a inside function:%d",a);

}

int main()

{

    printf("Before modification:%d",a);

    modify();

}
```

## 8. Using Augmented Assignment Operators

Question: Write a program that uses augmented assignment operators (+=, -=, *=, /=) to perform calculations on an integer variable initialized to 100. Print the value after each operation.

```c
#include<stdio.h>

int main()

{

    int a=100;

    a+=10;

    printf("+= :%d",a);

    a-=10;

    printf("\n-= :%d",a);

    a*=10;

    printf("\n*= :%d",a);

    a/=10;

    printf("\n/= :%d",a);

     return 0;

}
```

# 9. Array of Variables

Question: Create an array of integers with five elements. Initialize it with values of your choice, then write a program to calculate and print the sum of all elements in the array.

```c
#include<stdio.h>
int main()
{
    int sum=0;
    int arr[]={1,2,3,4,5};
    for(int i=0;i<5;i++)
    {
        sum=sum+arr[i];
    }
    printf("Sum is=%d",sum);
    return 0;
}
```

# 10.Assignment: User Authentication Program

## Objective

Create a C program that prompts the user for a username and password, then checks if the entered credentials match predefined values. Use logical operators to determine if the authentication is successful.

## Requirements

1. Define two constants for the correct username and password.
2. Prompt the user to enter their username and password.
3. Use logical operators (&&, ||, !) to check if:
4. If both are correct, display a success message.
5. Implement additional checks:
   - If the username is empty, display a message indicating that the username cannot be empty.
   - If the password is empty, display a message indicating that the password cannot be empty.
   - The username matches the predefined username AND the password matches the predefined password.
   - If either the username or password is incorrect, display an appropriate error message.

```c
#include<stdio.h>
```

```c
#include<string.h>

int main()
{

    char username[]="Nanda123";//correct
    char password[]="Nanda01";

    char user[18];
    char pass[20];

    printf("Username:");
    scanf("%s",&user);
    printf("\nPassword:");
    scanf("%s",&pass);

    if((strcmp(user,username)==0) && (strcmp(pass,password)==0))
    printf("Successfull login");
    else
    printf("Wrong user name or password");

    if((user[0]=="")||(pass[0]==""))
    printf("Empty space");
    return 0;
}
```

**11.Check if number is even or odd without using arithmetic operators.**

```c
int main()
{
int num;
scanf("%d",&num);
if((num&1)==0)
printf("even no");
else
print("odd no.");
}
```