

# **C PROGRAMMING**

## **PRACTICE PROGRAMS**

**NAME: S DEEPAK BHARATHWAJ**

**ROLL NO: CH.SC.U4CSE24015**

1. Single line comment

**CODE:**

```
1 //CH.SC.U4CSE24015  
2  
3 //This is a single line comment
```

2. Multi line comment

**CODE:**

```
1 //CH.SC.U4CSE24015  
2  
3 /*This is a multi line comment  
4 and it can be used in multiple lines*/  
5 |
```

3. Break and continue statements

**CODE:**

```
//CH.SC.U4CSE24015  
#include <stdio.h>  
int main() {  
    for(int i = 0 ; i<5 ; i++) {  
        if(i == 2){  
            continue;  
        }  
        if(i == 4){  
            break;  
        }  
    }  
    return 1;  
}
```

4. Goto – even and odd

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int num = 3;
    if (num % 2 == 0)
        goto even;
    else
        goto odd;
even:
    printf("Even number\n");
    return 1;
odd:
    printf("Odd number\n");
    return 1;
}
```

**OUTPUT:**

```
Odd number

Process returned 1 (0x1)    execution time : 0.080 s
Press any key to continue.
```

5. New line character

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    printf("Hello\nWorld");
    return 1;
}
```

**OUTPUT:**

```
Hello  
World  
Process returned 1 (0x1)    execution time : 0.381 s  
Press any key to continue.
```

6. Tab line character

**CODE:**

```
//CH.SC.U4CSE24015  
#include <stdio.h>  
int main() {  
    printf("Welcome \t to \t C");  
    return 1;  
}
```

**OUTPUT:**

```
Welcome          to          C  
Process returned 1 (0x1)    execution time : 0.342 s  
Press any key to continue.
```

7. Typedef program

**CODE:**

```
//CH.SC.U4CSE24015  
#include <stdio.h>  
typedef int hii;  
  
int main() {  
    hii age = 18;  
    printf("Age is : %i", age);  
    return 1;  
}
```

**OUTPUT:**

```
Age is : 18
Process returned 1 (0x1)    execution time : 0.462 s
Press any key to continue.
```

8. Input and display of datatypes

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int num1;
    float num2;
    char num3;
    printf("Enter int, float, and char : ");
    scanf("%d %f %c", &num1, &num2, &num3);
    printf("Values are : %d, %.2f, %c", num1, num2, num3);
    return 1;
}
```

**OUTPUT:**

```
Enter int, float, and char : 10 41.2421 D
Values are : 10, 41.24, D
Process returned 1 (0x1)    execution time : 7.134 s
Press any key to continue.
```

9. Display Without Input

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int num1 = 10;
    float num2 = 41.242;
    char num3 = 'D';
    printf("Int : %d, Float : %.2f, Char : %c", num1, num2, num3);
    return 1;
}
```

**OUTPUT:**

```
Int : 10, Float : 41.24, Char : D
Process returned 1 (0x1)    execution time : 0.421 s
Press any key to continue.
```

10. Sum of 2 Numbers with user input values

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int a, b, sum;
    printf("Enter the two numbers: ");
    scanf("%d %d", &a, &b);
    sum = a + b;
    printf("Sum is : %d", sum);
    return 1;
}
```

**OUTPUT:**

```
Enter the two numbers: 32
48
Sum is : 80
Process returned 1 (0x1)    execution time : 30.064 s
Press any key to continue.
```

## 11. Usage of Ternary Operator

### CODE:

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int a = 10, b = 20;
    int max = (a > b) ? a : b;
    printf("Greater : %d", max);
    return 1;
}
```

### OUTPUT:

```
Greater : 20
Process returned 1 (0x1)    execution time : 0.304 s
Press any key to continue.
```

## 12. To display a character

### CODE:

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    char ch = 'D';
    printf("Character : %c", ch);
    return 1;
}
```

### OUTPUT:

```
Character : D
Process returned 1 (0x1)    execution time : 0.345 s
Press any key to continue.
```

## 13. To display a string

### CODE:

```
| //CH.SC.U4CSE24015
| #include <stdio.h>
| int main() {
|     char str[] = "Hello World";
|     printf("%s", str);
|     return 1;
| }
```

**OUTPUT:**

```
Hello World
Process returned 1 (0x1)    execution time : 0.278 s
Press any key to continue.
```

14. To display a Group of Strings

**CODE:**

```
| //CH.SC.U4CSE24015
| #include <stdio.h>
| int main() {
|     char ch[200];
|     printf("Enter the string : ");
|     fgets(ch, sizeof(ch), stdin);
|     puts(ch);
|     return 1;
| }
```

**OUTPUT:**

```
Enter the string : Hello , this is deepak!
Hello , this is deepak!

Process returned 1 (0x1)    execution time : 12.059 s
Press any key to continue.
```

15. C program to check whether a number is positive or negative

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int num;
    printf("Enter a number: ");
    scanf("%d", &num);
    if(num > 0)
        printf("Positive\n");
    else if(num < 0)
        printf("Negative\n");
    else
        printf("Zero\n");
    return 1;
}
```

**OUTPUT:**

```
Enter a number: 59
Positive

Process returned 1 (0x1)    execution time : 3.988 s
Press any key to continue.
```

#### 16. Reverse an input number using recursion

##### CODE:

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int num , rev = 0, rem;
    printf("Enter the number : ");
    scanf("%d", &num);
    while(num != 0) {
        rem = num % 10;
        rev = rev*10 + rem;
        num/=10;
    }
    printf("Reversed number : %d\n", rev);
    return 1;
}
```

**OUTPUT:**

```
Enter the number : 153
Reversed number : 351

Process returned 1 (0x1)    execution time : 23.702 s
Press any key to continue.
```

17. Program to find greatest of three numbers

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int x, y, z;
    printf("Enter three numbers: ");
    scanf("%d %d %d", &x, &y, &z);
    if(x >= y && x >= z)
        printf("%d is the greatest\n", x);
    else if(y >= x && y >= z)
        printf("%d is the greatest\n", y);
    else
        printf("%d is the greatest\n", z);
    return 1;
}
```

**OUTPUT:**

```
Enter three numbers: 3
12
7
12 is the greatest

Process returned 1 (0x1)    execution time : 11.128 s
Press any key to continue.
```

18. C Program to print Fibonacci series in a given range

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int n, t1 = 0, t2 = 1, nextTerm;
    printf("Enter number of terms: ");
    scanf("%d", &n);
    printf("Fibonacci Series: ");
    for(int i = 1; i <= n; i++) {
        printf("%d ", t1);
        nextTerm = t1 + t2;
        t1 = t2;
        t2 = nextTerm;
    }
    printf("\n");
    return 1;
}
```

---

**OUTPUT:**

```
Enter number of terms: 9
Fibonacci Series: 0 1 1 2 3 5 8 13 21

Process returned 1 (0x1)  execution time : 13.069 s
Press any key to continue.
```

## 19. C Program to find factorial of a given number

### CODE:

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int num;
    long fact = 1;
    printf("Enter a number: ");
    scanf("%d", &num);
    for(int i = 1; i <= num; i++) {
        fact *= i;
    }
    printf("Factorial of %d = %ld\n", num, fact);
    return 1;
}
```

---

### OUTPUT:

```
Enter a number: 8
Factorial of 8 = 40320

Process returned 0 (0x0)  execution time : 4.437 s
Press any key to continue.
```

## 20. Find Prime numbers in a given range

### **CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int start, end;
    printf("Enter start and end : ");
    scanf("%d %d", &start, &end);
    printf("Prime numbers between %d and %d:\n", start, end);
    for(int num = start; num <= end; num++) {
        int Prime = 1;
        if(num <= 1)
            continue;
        for(int i = 2; i <= num/2; i++) {
            if(num % i == 0) {
                Prime = 0;
                break;
            }
        }
        if(Prime)
            printf("%d ", num);
    }
    printf("\n");
    return 1;
}
```

### **OUTPUT:**

```
Enter start and end : 1 10
Prime numbers between 1 and 10:
2 3 5 7

Process returned 1 (0x1)    execution time : 3.502 s
Press any key to continue.
|
```

## 21. C Program to check if given number is Armstrong or not

### **CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
#include <math.h>
int main() {
    int num, sum, temp, i = 0;
    int ans = 0;
    printf("Enter a number: ");
    scanf("%d", &num);
    sum = num;
    while (sum != 0) {
        sum /= 10;
        ++i;
    }
    sum = num;
    while (sum != 0) {
        temp = sum % 10;
        ans += pow(temp, i);
        sum /= 10;
    }
    if(ans == num)
        printf("%d is an Armstrong number.\n", num);
    else
        printf("%d is not an Armstrong number.\n", num);
    return 1;
}
```

### **OUTPUT:**

```
Enter a number: 153
153 is an Armstrong number.
```

```
Process returned 1 (0x1)    execution time : 1.894 s
Press any key to continue.
```

## 22. C Program to check if given number is palindrome or not

### **CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int num, rev = 0, temp, sum;
    printf("Enter a number: ");
    scanf("%d", &num);
    sum = num;
    while(num != 0) {
        temp = num % 10;
        rev = rev * 10 + temp;
        num /= 10;
    }
    if(sum == rev)
        printf("%d is a palindrome.\n", sum);
    else
        printf("%d is not a palindrome.\n", sum);
    return 1;
}
```

### **OUTPUT:**

```
Enter a number: 121
121 is a palindrome.

Process returned 1 (0x1)  execution time : 3.015 s
Press any key to continue.
```

## 23. C Program to display palindrome numbers in a given range

### **CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int start, end;
    printf("Enter start and end : ");
    scanf("%d %d", &start, &end);
    printf("Palindrome numbers between %d and %d:\n", start, end);
    for(int num = start; num <= end; num++) {
        int i = num, rev = 0, temp;
        while(i != 0) {
            temp = i % 10;
            rev = rev * 10 + temp;
            i /= 10;
        }
        if(rev == num)
            printf("%d ", num);
    }
    printf("\n");
    return 1;
}
```

---

### **OUTPUT:**

```
Enter start and end : 100 300
Palindrome numbers between 100 and 300:
101 111 121 131 141 151 161 171 181 191 202 212 222 232 242 252 262 272 282 292

Process returned 1 (0x1)  execution time : 3.179 s
Press any key to continue.
```

## 24. C Program to find out the ASCII value of a character

### CODE:

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    char ch;
    printf("Enter the character: ");
    scanf(" %c", &ch);
    printf("ASCII value is : %d\n", ch, ch);
    return 1;
}
```

---

### OUTPUT:

```
Enter the character: A
ASCII value is : 65

Process returned 1 (0x1)    execution time : 2.108 s
Press any key to continue.
```

## 25. C Program to find the size of int, float, double and char

### CODE:

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    printf("Size of int : %zu bytes\n", sizeof(int));
    printf("Size of float : %zu bytes\n", sizeof(float));
    printf("Size of double : %zu bytes\n", sizeof(double));
    printf("Size of char : %zu bytes\n", sizeof(char));
    return 1;
}
```

### OUTPUT:

```
Size of int : 4 bytes
Size of float : 4 bytes
Size of double : 8 bytes
Size of char : 1 bytes

Process returned 1 (0x1)    execution time : 0.562 s
Press any key to continue.
```

## 26. C Program to find sum of first n natural numbers

### CODE:

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int num, sum = 0;
    printf("Enter the number : ");
    scanf("%d", &num);
    sum = num * (num + 1) / 2;
    printf("Sum of first %d natural numbers = %d\n", num, sum);
    return 1;
}
```

### OUTPUT:

```
Enter the number : 10
Sum of first 10 natural numbers = 55

Process returned 1 (0x1)    execution time : 2.925 s
Press any key to continue.
```

## 27. C Program to print integer entered by user

### CODE:

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int num;
    printf("Enter an integer : ");
    scanf("%d", &num);
    printf("You entered : %d\n", num);
    return 1;
}
```

---

### OUTPUT:

```
Enter an integer : 15
You entered : 15

Process returned 1 (0x1)    execution time : 0.983 s
Press any key to continue.
```

## 28. Occurrence of element in an array

### CODE:

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main(){
    int n,k,i,count=0;
    printf("Enter the size of the array : ");
    scanf("%d",&n);
    int arr[n];
    printf("Enter the elements of array : ");
    for(i=0;i<n;i++){
        scanf("%d",&arr[i]);
    }
    printf("Enter the number to know it's count : ");
    scanf("%d",&k);
    for(i=0;i<n;i++){
        if(arr[i]==k){
            count++;
        }
    }
    printf("%d occurs %d times",k,count);
    return 1;
}
```

**OUTPUT:**

```
Enter the size of the array : 5
Enter the elements of array : 1 2 2 4 5
Enter the number to know it's count : 2
2 occurs 2 times
-----
Process exited after 6.896 seconds with return value 1
Press any key to continue . . .
```

29.Greatest of 3 numbers

**CODE:**

```
//ch.sc.u4cse24015
#include <stdio.h>
int main(){
    int a, b, c;
    printf("Enter 3 numbers: ");
    scanf("%d %d %d", &a, &b, &c);
    if (a >= b && a >= c)
        printf("Greatest: %d\n", a);
    else if (b >= a && b >= c)
        printf("Greatest: %d\n", b);
    else
        printf("Greatest: %d\n", c);
    return 0;
}
```

**OUTPUT:**

```
Enter 3 numbers: 5 86 32
Greatest: 86
-----
Process exited after 10.63 seconds with return value 0
Press any key to continue . . . |
```

### 30.Prime numbers in an array

#### **CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int isPrime(int num) {
    int i;
    if (num <= 1) return 0;
    for(i = 2; i <= num / 2; i++) {
        if (num % i == 0)
            return 0;
    }
    return 1;
}
int main() {
    int n, i, num;
    printf("Enter the size of the array: ");
    scanf("%d", &n);
    int input[n];
    int count = 0;
    printf("Enter the elements of the array: ");
    for (i = 0; i < n; i++) {
        scanf("%d", &input[i]);
        if (isPrime(input[i])) {
            count++;
        }
    }
    if (count == 0) {
        printf("No prime numbers found.\n");
        return 0;
    }
    int prime[count];
    int j = 0;
    for (i = 0; i < n; i++) {
        if (isPrime(input[i])) {
            prime[j++] = input[i];
        }
    }
    printf("Prime numbers are: ");
    for (i = 0; i < count; i++) {
        printf("%d ", prime[i]);
    }
    return 0;
}
```

#### **OUTPUT:**

```
Enter the size of the array: 5
Enter the elements of the array: 3 6 9 1 4
Prime numbers are: 3
-----
Process exited after 11.54 seconds with return value 0
Press any key to continue . . . |
```

### **31.Reverse of a number**

#### **CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main(){
    int num,rem,rev_num=0;
    printf("Enter the number: ");
    scanf("%d",&num);
    while(num!=0){
        rem=num%10;
        rev_num=rev_num*10+rem;
        num/=10;
    }
    printf("Reversed number: %d",rev_num);
    return 0;
}
```

#### **OUTPUT:**

```
Enter the number: 24022
Reversed number: 22042
-----
Process exited after 18.27 seconds with return value 0
Press any key to continue . . . |
```

### **32.Sum of the terms x+3x+5x+...**

#### **CODE:**

```
//SUM=x+3x+5x+7x+.....+n terms
//CH.SC.U4CSE24015
#include <stdio.h>
int main(){
    int n,x,i,pro=0,sum;
    printf("Enter the value of n and x: ");
    scanf("%d",&n);
    scanf("%d",&x);
    for(i=1;i<n*2;i++){
        if(i%2!=0){
            pro=pro+i;
        }
        else{
            continue;
        }
    }
    sum=x*pro;
    printf("%d",sum);
}
```

**OUTPUT:**

```
Enter the value of n and x: 5 2
50
-----
Process exited after 1.127 seconds with return value 2
Press any key to continue . . . |
```

33. C program to update two variables i,j using for loop for 5 times with  $i=i+1, j=j+2$ .

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int i = 0, j = 0, k;
    for(k = 0; k < 5; k++) {
        i = i + 1;
        j = j + 2;
        printf("i = %d, j = %d\n", i, j);
    }
    return 0;
}
```

**OUTPUT:**

```
i = 1, j = 2
i = 2, j = 4
i = 3, j = 6
i = 4, j = 8
i = 5, j = 10
-----
Process exited after 0.1839 seconds with return value 0
Press any key to continue . . . |
```

34. C program to demonstrate that a variable declared locally within a for loop does not affect a globally declared variable of the same name.

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int i = 100;
int main() {
    int i;
    for (i = 0; i < 3; i++) {
        printf("Inside loop i = %d\n", i);
    }
    printf("Outside loop i = %d\n", i);
    return 0;
}
```

**OUTPUT:**

```
Inside loop i = 0
Inside loop i = 1
Inside loop i = 2
Outside loop i = 3
-----
Process exited after 0.1998 seconds with return value 0
Press any key to continue . . . |
```

35. Write a C program to print the multiplication tables from 1 to n, each up to 10, based on user input.

**CODE:**

```
//CH.SC.U4CSE24025
#include <stdio.h>
int main() {
    int n,i,j;
    printf("Enter value of n: ");
    scanf("%d", &n);
    for (i = 1; i <= n; i++) {
        printf("Table of %d\t", i);
    }
    printf("\n");
    for (j = 1; j <= 10; j++) {
        for (i = 1; i <= n; i++) {
            printf("%d x %d = %d\t", i, j, i * j);
        }
        printf("\n");
    }
    return 0;
}
```

## **OUTPUT:**

```
Enter value of n: 9
Table of 1      Table of 2      Table of 3      Table of 4      Table of 5      Table of 6      Table of 7      Table of 8      Table of 9
1 x 1 = 1      2 x 1 = 2      3 x 1 = 3      4 x 1 = 4      5 x 1 = 5      6 x 1 = 6      7 x 1 = 7      8 x 1 = 8      9 x 1 = 9
1 x 2 = 2      2 x 2 = 4      3 x 2 = 6      4 x 2 = 8      5 x 2 = 10     6 x 2 = 12     7 x 2 = 14     8 x 2 = 16     9 x 2 = 18
1 x 3 = 3      2 x 3 = 6      3 x 3 = 9      4 x 3 = 12     5 x 3 = 15     6 x 3 = 18     7 x 3 = 21     8 x 3 = 24     9 x 3 = 27
1 x 4 = 4      2 x 4 = 8      3 x 4 = 12     4 x 4 = 16     5 x 4 = 20     6 x 4 = 24     7 x 4 = 28     8 x 4 = 32     9 x 4 = 36
1 x 5 = 5      2 x 5 = 10     3 x 5 = 15     4 x 5 = 20     5 x 5 = 25     6 x 5 = 30     7 x 5 = 35     8 x 5 = 40     9 x 5 = 45
1 x 6 = 6      2 x 6 = 12     3 x 6 = 18     4 x 6 = 24     5 x 6 = 30     6 x 6 = 36     7 x 6 = 42     8 x 6 = 48     9 x 6 = 54
1 x 7 = 7      2 x 7 = 14     3 x 7 = 21     4 x 7 = 28     5 x 7 = 35     6 x 7 = 42     7 x 7 = 49     8 x 7 = 56     9 x 7 = 63
1 x 8 = 8      2 x 8 = 16     3 x 8 = 24     4 x 8 = 32     5 x 8 = 40     6 x 8 = 48     7 x 8 = 56     8 x 8 = 64     9 x 8 = 72
1 x 9 = 9      2 x 9 = 18     3 x 9 = 27     4 x 9 = 36     5 x 9 = 45     6 x 9 = 54     7 x 9 = 63     8 x 9 = 72     9 x 9 = 81
1 x 10 = 10     2 x 10 = 20    3 x 10 = 30    4 x 10 = 40    5 x 10 = 50    6 x 10 = 60    7 x 10 = 70    8 x 10 = 80    9 x 10 = 90

-----
Process exited after 1.072 seconds with return value 0
Press any key to continue . . . |
```

## **36. Pattern Printing:**

\*

\*\*

\*\*\*

## **CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int i,j;
    for (i = 1; i <= 3; i++) {
        for (j = 1; j <= i; j++) {
            printf("*");
        }
        printf("\n");
    }
    return 0;
}
```

## **OUTPUT:**

```
*
**
***
```

```
-----
Process exited after 0.2259 seconds with return value 0
Press any key to continue . . . |
```

37. Pattern Printing:

1

23

456

**CODE:**

```
///CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int i,j;
    int num = 1;
    for (i = 1; i <= 3; i++) {
        for (j = 1; j <= i; j++) {
            printf("%d", num++);
        }
        printf("\n");
    }
    return 0;
}
```

**OUTPUT:**

```
1
23
456
```

```
-----
Process exited after 0.1675 seconds with return value 0
Press any key to continue . . . |
```

38. Pattern Printing:

```
1  
11  
111  
1111
```

**CODE:**

```
//CH.SC.U4CSE24025  
#include <stdio.h>  
int main() {  
    int i,j;  
    for (i = 1; i <= 4; i++) {  
        for (j = 1; j <= i; j++) {  
            printf("1");  
        }  
        printf("\n");  
    }  
    return 0;  
}
```

**OUTPUT:**

```
1  
11  
111  
1111  
-----  
Process exited after 0.1877 seconds with return value 0  
Press any key to continue . . . |
```

39. Pattern Printing:

```
1  
2 3  
4 5 6 7
```

**CODE:**

```
//CH.SC.U4CSE24015  
#include <stdio.h>  
int main() {  
    int i,j,num = 1;  
    for (i = 1; i <= 3; i++) {  
        for (j = 1; j <= i * 2 - 1; j++) {  
            printf("%d ", num++);  
        }  
        printf("\n");  
    }  
    return 0;  
}
```

**OUTPUT:**

```
1  
2 3 4  
5 6 7 8 9
```

```
-----  
Process exited after 0.4001 seconds with return value 0  
Press any key to continue . . .
```

40. C program that prints "hello" repeatedly in an infinite loop only when the character variable ch is set to 'y'. Initialize ch with any value. You may use a while loop to implement the logic.

## CODE:

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    char ch = 'y';
    while (ch == 'y') {
        printf("Hello\n");
    }
    return 0;
}
```

## **OUTPUT:**

41. C program to print the numbers from 1 to 10 using a while loop.

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int i = 1;
    while (i <= 10) {
        printf("%d\n", i);
        i++;
    }
    return 0;
}
```

**OUTPUT:**

```
1
2
3
4
5
6
7
8
9
10
```

```
-----
Process exited after 0.2942 seconds with return value 0
Press any key to continue . . .
```

42. Use #define MAX in array program using break continue and goto also.

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
#define MAX 5
int main() {
    int i, arr[MAX] = {1, 2, 0, 4, 5};
    for (i = 0; i < MAX; i++) {
        if (arr[i] == 0)
            continue;
        if (arr[i] == 4)
            goto end;
        printf("%d ", arr[i]);
    }
end:
    printf("\nExited loop using goto.\n");
    return 0;
}
```

**OUTPUT:**

```
1 2
Exited loop using goto.

-----
Process exited after 0.27 seconds with return value 0
Press any key to continue . . .
```

43. Pre-define an Array and print it

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int i;
    int arr[] = {5, 10, 15, 20};
    int size = sizeof(arr) / sizeof(arr[0]);
    for (i = 0; i < size; i++) {
        printf("%d ", arr[i]);
    }
    return 0;
}
```

**OUTPUT:**

```
5 10 15 20
-----
Process exited after 0.4219 seconds with return value 0
Press any key to continue . . .
```

44. Get input of array from user and print it

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int i, arr[5];
    printf("Enter 5 elements:\n");
    for (i = 0; i < 5; i++) {
        scanf("%d", &arr[i]);
    }
    printf("You entered:\n");
    for (i = 0; i < 5; i++) {
        printf("%d ", arr[i]);
    }
    return 0;
}
```

**OUTPUT:**

```
Enter 5 elements:
1 2 3 4 5
You entered:
1 2 3 4 5
-----
Process exited after 3.257 seconds with return value 0
Press any key to continue . . .
```

45. Print only three element of array and then break the loop

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int i;
    int arr[5] = {10, 20, 30, 40, 50};
    for (i = 0; i < 5; i++) {
        printf("%d ", arr[i]);
        if (i == 2) break;
    }
    return 0;
}
```

**OUTPUT:**

```
10 20 30
```

```
-----  
Process exited after 2.097 seconds with return value 0  
Press any key to continue . . . |
```

46. C program to find the largest number in array:{10,12,16,4,8}

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int i;
    int arr[] = {10, 12, 16, 4, 8};
    int max = arr[0];
    for (i = 1; i < 5; i++) {
        if (arr[i] > max)
            max = arr[i];
    }
    printf("Largest = %d\n", max);
    return 0;
}
```

**OUTPUT:**

```
Largest = 16  
-----  
Process exited after 0.9183 seconds with return value 0  
Press any key to continue . . . |
```

47. C Program to find the second largest element in array:{10,8,16,4,12}

**CODE:**

```
//CH.SC.U4CSE24015  
#include <stdio.h>  
int main() {  
    int i;  
    int arr[] = {10, 8, 16, 4, 12};  
    int first = arr[0], second = -1;  
    for (i = 1; i < 5; i++) {  
        if (arr[i] > first) {  
            second = first;  
            first = arr[i];  
        } else if (arr[i] > second && arr[i] != first) {  
            second = arr[i];  
        }  
    }  
    printf("Second largest = %d\n", second);  
    return 0;  
}
```

**OUTPUT:**

```
Second largest = 12  
-----  
Process exited after 0.2924 seconds with return value 0  
Press any key to continue . . . |
```

48. C program to print the numbers from 0 to n using array.

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int n,i;
    printf("Enter n: ");
    scanf("%d", &n);
    int arr[n];
    for (i = 0; i <= n; i++) {
        arr[i] = i;
    }
    printf("Numbers from 0 to %d:\n", n);
    for (i = 0; i <= n; i++) {
        printf("%d ", arr[i]);
    }
    return 0;
}
```

**OUTPUT:**

```
Enter n: 5
Numbers from 0 to 5:
0 1 2 3 4 5
-----
Process exited after 1.679 seconds with return value 0
Press any key to continue . . . |
```

49. C program to reverse a number using array

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int num,j,i = 0;
    int digits[10];
    printf("Enter a number: ");
    scanf("%d", &num);
    while (num != 0) {
        digits[i++] = num % 10;
        num /= 10;
    }
    printf("Reversed number: ");
    for (j = 0; j < i; j++) {
        printf("%d", digits[j]);
    }
    return 0;
}
```

**OUTPUT:**

```
Enter a number: 82
Reversed number: 28
-----
Process exited after 2.575 seconds with return value 0
Press any key to continue . . . |
```

50. C program to create a 2-dimensional array and it should print output:

1 2

3 4

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int i,j;
    int arr[2][2] = {{1, 2}, {3, 4}};
    for (i = 0; i < 2; i++) {
        for (j = 0; j < 2; j++) {
            printf("%d ", arr[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```

**OUTPUT:**

```
1 2
3 4
```

```
-----
Process exited after 0.2419 seconds with return value 0
Press any key to continue . . . |
```

51. Program for printing the sum of column and sum of rows for the array:

|8 3 9 0 10|

|3 5 17 1 1|

|2 8 6 23 1|

|15 7 3 2 9|

|6 14 2 6 0|

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int i,j;
    int arr[5][5] = {
        {8, 3, 9, 0, 10},
        {3, 5, 17, 1, 1},
        {2, 8, 6, 23, 1},
        {15, 7, 3, 2, 9},
        {6, 14, 2, 6, 0}
    };
    for (i = 0; i < 5; i++) {
        int rowSum = 0;
        for (j = 0; j < 5; j++) {
            rowSum += arr[i][j];
        }
        printf("Sum of row %d = %d\n", i + 1, rowSum);
    }
    for (j = 0; j < 5; j++) {
        int colSum = 0;
        for (i = 0; i < 5; i++) {
            colSum += arr[i][j];
        }
        printf("Sum of column %d = %d\n", j + 1, colSum);
    }
    return 0;
}
```

**OUTPUT:**

```
Sum of row 1 = 30
Sum of row 2 = 27
Sum of row 3 = 40
Sum of row 4 = 36
Sum of row 5 = 28
Sum of column 1 = 34
Sum of column 2 = 37
Sum of column 3 = 37
Sum of column 4 = 32
Sum of column 5 = 21
-----
Process exited after 0.1839 seconds with return value 0
Press any key to continue . . . |
```

## 52. C Program to perform matrix multiplication

### **CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int i,j,k;
    int a[2][2] = {{1, 2}, {3, 4}};
    int b[2][2] = {{5, 6}, {7, 8}};
    int c[2][2] = {0};

    for (i = 0; i < 2; i++) {
        for (j = 0; j < 2; j++) {
            for (k = 0; k < 2; k++) {
                c[i][j] += a[i][k] * b[k][j];
            }
        }
    }

    printf("Matrix multiplication result:\n");
    for (i = 0; i < 2; i++) {
        for (j = 0; j < 2; j++) {
            printf("%d ", c[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```

### **OUTPUT:**

```
Matrix multiplication result:
19 22
43 50
-----
Process exited after 0.1917 seconds with return value 0
Press any key to continue . . . |
```

53. C Program to convert upper case to lower case

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
#include <ctype.h>
int main() {
    char ch;
    printf("Enter an uppercase letter: ");
    scanf(" %c", &ch);
    printf("Lowercase: %c\n", tolower(ch));
    return 0;
}
```

**OUTPUT:**

```
Enter an uppercase letter: D
Lowercase: d

-----
Process exited after 2.844 seconds with return value 0
Press any key to continue . . . |
```

54. C Program to calculate the sum of first n terms of the series:

$x+3x+5x+\dots+n$ (Hint:x(1+3+5+...n odd number))

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int x, i, n, sum = 0;
    printf("Enter value of x and number of terms: ");
    scanf("%d %d", &x, &n);
    for (i = 1; i <= 2 * n - 1; i += 2) {
        sum += i * x;
    }
    printf("Sum of series = %d\n", sum);
    return 0;
}
```

**OUTPUT:**

```
Enter value of x and number of terms: 2 5
Sum of series = 50
-----
Process exited after 2.964 seconds with return value 0
Press any key to continue . . . |
```

55. C Program to check frequency of a given number in an array.

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int i;
    int arr[10], n, count = 0;
    printf("Enter 10 numbers:\n");
    for (i = 0; i < 10; i++) {
        scanf("%d", &arr[i]);
    }
    printf("Enter number to find frequency: ");
    scanf("%d", &n);

    for (i = 0; i < 10; i++) {
        if (arr[i] == n)
            count++;
    }
    printf("Frequency of %d = %d\n", n, count);
    return 0;
}
```

**OUTPUT:**

```
Enter 10 numbers:
1 2 3 4 5 6 7 8 9 10
Enter number to find frequency: 5
Frequency of 5 = 1
-----
Process exited after 6.359 seconds with return value 0
Press any key to continue . . . |
```

56. Write a program to get an multi-dimensional array from user for 5 students with 3 subject for each of them and print the topper and his avg mark

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int i,j;
    int marks[5][3];
    int total[5] = {0}, max = 0, topper = 0;
    printf("Enter marks of 5 students (3 subjects each):\n");
    for (i = 0; i < 5; i++) {
        printf("Student %d:\n", i + 1);
        for (j = 0; j < 3; j++) {
            scanf("%d", &marks[i][j]);
            total[i] += marks[i][j];
        }
    }
    for (i = 0; i < 5; i++) {
        if (total[i] > max) {
            max = total[i];
            topper = i;
        }
    }
    printf("Topper: Student %d with average %.2f\n", topper + 1, max / 3.0);
    return 0;
}
```

**OUTPUT:**

```
Enter marks of 5 students (3 subjects each):
Student 1:
98
89
75
Student 2:
99
100
89
Student 3:
24
76
56
Student 4:
97
25
85
Student 5:
90
86
77
Topper: Student 2 with average 96.00
-----
Process exited after 23.2 seconds with return value 0
Press any key to continue . . . |
```

57. A multiplex wants to record ratings given by 4 viewers for 3 movies.

- Use a 2D array to store ratings (out of 10).
- Use a 1D array to store the average rating of each movie.
- Display:
  - Ratings given by each viewer
  - Average rating of each movie
  - Movie with the highest average rating.

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int i,j;
    int ratings[4][3];
    float avg[3] = {0};
    int maxIndex = 0;
    printf("Enter ratings by 4 viewers for 3 movies (out of 10):\n");
    for (i = 0; i < 4; i++) {
        printf("Viewer %d:\n", i + 1);
        for (j = 0; j < 3; j++) {
            scanf("%d", &ratings[i][j]);
            avg[j] += ratings[i][j];
        }
    }
    for (i = 0; i < 3; i++) {
        avg[i] /= 4.0;
    }
    for (i = 1; i < 3; i++) {
        if (avg[i] > avg[maxIndex])
            maxIndex = i;
    }
    printf("\nAverage Ratings:\n");
    for (i = 0; i < 3; i++) {
        printf("Movie %d: %.2f\n", i + 1, avg[i]);
    }
    printf("Highest rated movie: Movie %d\n", maxIndex + 1);
    return 0;
}
```

**OUTPUT:**

```
Enter ratings by 4 viewers for 3 movies (out of 10):
```

```
Viewer 1:
```

```
8
```

```
9
```

```
7
```

```
Viewer 2:
```

```
9
```

```
9
```

```
10
```

```
Viewer 3:
```

```
7
```

```
9
```

```
8
```

```
Viewer 4:
```

```
9
```

```
8
```

```
8
```

```
Average Ratings:
```

```
Movie 1: 8.25
```

```
Movie 2: 8.75
```

```
Movie 3: 8.25
```

```
Highest rated movie: Movie 2
```

---

```
-----
```

```
Process exited after 12.2 seconds with return value 0
```

```
Press any key to continue . . . |
```

## 58. Cricket Team Scorecard

A cricket team played 5 matches. For each match, it has 11 players' scores.

- Use a 2D array to store scores.
- Use a 1D array to store the total score of each match.
- Display:
  - All scores of players match-wise
  - Total score for each match
  - Match with the highest total score.

### **CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>

int main() {
    int i,j;
    int score[5][11];
    int total[5] = {0}, max = 0, match = 0;

    for (i = 0; i < 5; i++) {
        printf("Enter scores for match %d:\n", i + 1);
        for (j = 0; j < 11; j++) {
            scanf("%d", &score[i][j]);
            total[i] += score[i][j];
        }
    }
    for (i = 0; i < 5; i++) {
        if (total[i] > max) {
            max = total[i];
            match = i;
        }
    }
    printf("Total scores:\n");
    for (i = 0; i < 5; i++) {
        printf("Match %d: %d\n", i + 1, total[i]);
    }
    printf("Match with highest score: Match %d\n", match + 1);
    return 0;
}
```

**OUTPUT:**

```
Enter scores for match 1:  
98 73 35 86 25 24 27 82 88 97 100  
Enter scores for match 2:  
80 70 74 02 84 94 74 94 84 12 90  
Enter scores for match 3:  
97 68 64 83 02 78 98 62 45 11 01  
Enter scores for match 4:  
97 100 24 98 43 78 97 24 87 52 52  
Enter scores for match 5:  
87 97 25 84 82 12 53 72 84 27 93  
Total scores:  
Match 1: 735  
Match 2: 758  
Match 3: 609  
Match 4: 752  
Match 5: 716  
Match with highest score: Match 2
```

---

```
Process exited after 90.98 seconds with return value 0  
Press any key to continue . . . |
```

59.

Record daily temperature of 7 days at 2 different times( Morning , Afternoon and Evening)

- Use a 2D array to store temperatures recorded per day at different times.
- Use a 1D array to store the average temperature for each day.
- Display:
  - All temperature values recorded day-wise.
  - Average temperature for each day.
  - The day with the highest average temperature.

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>

int main() {
    int i,j;
    float temp[7][3], avg[7], maxAvg = 0;
    int day = 0;
    for (i = 0; i < 7; i++) {
        printf("Enter temps for Day %d (Morning, Afternoon, Evening):\n", i + 1);
        float sum = 0;
        for (j = 0; j < 3; j++) {
            scanf("%f", &temp[i][j]);
            sum += temp[i][j];
        }
        avg[i] = sum / 3;
        if (avg[i] > maxAvg) {
            maxAvg = avg[i];
            day = i;
        }
    }
    printf("Average temperatures:\n");
    for (i = 0; i < 7; i++) {
        printf("Day %d: %.2f\n", i + 1, avg[i]);
    }
    printf("Hottest day: Day %d\n", day + 1);
    return 0;
}
```

**OUTPUT:**

```
Enter temps for Day 1 (Morning, Afternoon, Evening):
192.34 92.59 99.82
Enter temps for Day 2 (Morning, Afternoon, Evening):
98.445 32.34 90.392
Enter temps for Day 3 (Morning, Afternoon, Evening):
94.95 78.84 81.83
Enter temps for Day 4 (Morning, Afternoon, Evening):
98.35 93.9 78.8
Enter temps for Day 5 (Morning, Afternoon, Evening):
24.5 67.89 98.34
Enter temps for Day 6 (Morning, Afternoon, Evening):
85.7 25.6 68.9
Enter temps for Day 7 (Morning, Afternoon, Evening):
68.9 12.5 100.83
Average temperatures:
Day 1: 128.25
Day 2: 73.73
Day 3: 85.21
Day 4: 90.35
Day 5: 63.58
Day 6: 60.07
Day 7: 60.74
Hottest day: Day 1

-----
Process exited after 83.42 seconds with return value 0
Press any key to continue . . . |
```

60. A library has 5 shelves, and each shelf contains 4 different books.

- Use a 2D array to store the number of copies of each book on every shelf.
- Use a 1D array to store the total number of books on each shelf.
- Display:
  - Copies of books shelf-wise.
  - Total books on each shelf.
  - Shelf with the maximum number of books.

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int i,j;
    int books[5][4], total[5] = {0}, max = 0, shelf = 0;
    for (i = 0; i < 5; i++) {
        printf("Enter copies for Shelf %d (4 books):\n", i + 1);
        for (j = 0; j < 4; j++) {
            scanf("%d", &books[i][j]);
            total[i] += books[i][j];
        }
    }
    for (i = 0; i < 5; i++) {
        if (total[i] > max) {
            max = total[i];
            shelf = i;
        }
    }
    printf("Total books per shelf:\n");
    for (i = 0; i < 5; i++) {
        printf("Shelf %d: %d\n", i + 1, total[i]);
    }
    printf("Shelf with most books: Shelf %d\n", shelf + 1);
    return 0;
}
```

**OUTPUT:**

```
Enter copies for Shelf 1 (4 books):  
2 7 9 3  
Enter copies for Shelf 2 (4 books):  
7 2 1 1  
Enter copies for Shelf 3 (4 books):  
3 4 5 1  
Enter copies for Shelf 4 (4 books):  
9 10 7 6  
Enter copies for Shelf 5 (4 books):  
1 2 4 5  
Total books per shelf:  
Shelf 1: 21  
Shelf 2: 11  
Shelf 3: 13  
Shelf 4: 32  
Shelf 5: 12  
Shelf with most books: Shelf 4
```

---

```
-----  
Process exited after 16.75 seconds with return value 0  
Press any key to continue . . . |
```

61. In a sports event, 6 players participate in 4 games.

Write a C program to:

- Use a 2D array to store scores of each player for each game.
- Use a 1D array to store the total score of each player.
- Display:
  - Player-wise scores
  - Total score of each player
  - Player with the highest total score

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int i,j;
    int score[6][4], total[6] = {0}, max = 0, topPlayer = 0;
    for (i = 0; i < 6; i++) {
        printf("Enter scores for Player %d (4 games):\n", i + 1);
        for (j = 0; j < 4; j++) {
            scanf("%d", &score[i][j]);
            total[i] += score[i][j];
        }
    }
    for (i = 0; i < 6; i++) {
        if (total[i] > max) {
            max = total[i];
            topPlayer = i;
        }
    }
    printf("Total scores per player:\n");
    for (i = 0; i < 6; i++) {
        printf("Player %d: %d\n", i + 1, total[i]);
    }
    printf("Top scorer: Player %d\n", topPlayer + 1);
    return 0;
}
```

**OUTPUT:**

```
Enter scores for Player 1 (4 games):
75 29 98 79
Enter scores for Player 2 (4 games):
75 98 23 93
Enter scores for Player 3 (4 games):
100 97 72 67
Enter scores for Player 4 (4 games):
52 87 100 87
Enter scores for Player 5 (4 games):
27 98 78 99 88
Enter scores for Player 6 (4 games):
76 87 82 99
Total scores per player:
Player 1: 281
Player 2: 289
Player 3: 336
Player 4: 326
Player 5: 302
Player 6: 333
Top scorer: Player 3

-----
Process exited after 36.82 seconds with return value 0
Press any key to continue . . . |
```

62. C - program which gets an array as an input and remove the duplicate elements from the array and print the new array.

**CODE:**

---

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int arr[100], n, i, j, k;
    printf("Enter number of elements: ");
    scanf("%d", &n);
    printf("Enter elements:\n");
    for (i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
    }
    for (i = 0; i < n; i++) {
        for (j = i + 1; j < n; ) {
            if (arr[i] == arr[j]) {
                for (k = j; k < n - 1; k++) {
                    arr[k] = arr[k + 1];
                }
                n--;
            } else {
                j++;
            }
        }
    }
    printf("Array after removing duplicates:\n");
    for (i = 0; i < n; i++) {
        printf("%d ", arr[i]);
    }
    return 0;
}
```

**OUTPUT:**

```
Enter number of elements: 5
Enter elements:
8 0 7 6 6
Array after removing duplicates:
8 0 7 6
-----
Process exited after 8.404 seconds with return value 0
Press any key to continue . . . |
```

63. Program for Jumping out of nested loop using goto

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int i,j;
    for (i = 1; i <= 3; i++) {
        for (j = 1; j <= 3; j++) {
            if (i == 2 && j == 2)
                goto exit;
            printf("i=%d j=%d\n", i, j);
        }
    }
exit:
    printf("Exited nested loop using goto\n");
    return 0;
}
```

**OUTPUT:**

```
i=1 j=1
i=1 j=2
i=1 j=3
i=2 j=1
Exited nested loop using goto
-----
Process exited after 0.2055 seconds with return value 0
Press any key to continue . . . |
```

64. Check odd/even using ternary

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int num;
    printf("Enter a number: ");
    scanf("%d", &num);
    (num % 2 == 0) ? printf("Even\n") : printf("Odd\n");
    return 0;
}
```

**OUTPUT:**

```
Enter a number: 5
Odd
-----
Process exited after 0.9746 seconds with return value 0
Press any key to continue . . . |
```

65. Eligibility to vote using ternary

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int age;
    printf("Enter your age: ");
    scanf("%d", &age);
    (age >= 18) ? printf("Eligible to vote\n") : printf("Not eligible to vote\n");
    return 0;
}
```

**OUTPUT:**

```
Enter your age: 18
Eligible to vote

-----
Process exited after 1.215 seconds with return value 0
Press any key to continue . . . |
```

66. Arithmetic operations

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int a, b;
    printf("Enter two numbers: ");
    scanf("%d %d", &a, &b);
    printf("Addition: %d\n", a + b);
    printf("Subtraction: %d\n", a - b);
    printf("Multiplication: %d\n", a * b);
    if (b != 0)
        printf("Division: %d\n", a / b);
    else
        printf("Division by zero not allowed\n");
    return 0;
}
```

**OUTPUT:**

```
Enter two numbers: 22
15
Addition: 37
Subtraction: 7
Multiplication: 330
Division: 1
```

```
-----
Process exited after 4.214 seconds with return value 0
Press any key to continue . . . |
```

67. Pre and post increment in all ways

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    int a = 5, b = 5;
    printf("a++ = %d\n", a++);
    printf("Now a = %d\n", a);
    printf("++b = %d\n", ++b);
    printf("Now b = %d\n", b);
    return 0;
}
```

**OUTPUT:**

```
a++ = 5
Now a = 6
++b = 6
Now b = 6
```

```
-----
Process exited after 0.2009 seconds with return value 0
Press any key to continue . . . |
```

68. Print sentence with fgets limit 10

**CODE:**

```
//CH.SC.U4CSE24015
#include <stdio.h>
int main() {
    char input[10];
    printf("Enter a sentence (max 9 characters): ");
    fgets(input, sizeof(input), stdin);
    printf("You entered: %s", input);
    return 0;
}
```

**OUTPUT:**

```
Enter a sentence (max 9 characters): hello welcome to C
You entered: hello wel
-----
```

```
Process exited after 6.06 seconds with return value 0
Press any key to continue . . . |
```