ARAVALI COLLEGE OF ENGINEERING & MANAGEMENT

Jasana Tigoan Road Greater Faridabad Haryana, 121006

VAC C LAB FILE

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

(2024-2025)

FACULTY INCHARGE

Mr.Pradeep Sir

(Assistant Professor)

Submitte By:

Name: Ankit kumar

Roll No: 24011312010

Index

13	Write a program to find the factorial of a number using functions.		
14	Write a program to swap two numbers by using call by value and call by reference		
15	Write a program to illustrate how to read a string from terminal.		
16	Write a program to print sum of two matrices.		
17	Write a program to print multiply of two matrices.		
18	Write a program to show address of variables by using pointers		
19	Write a program for linear searching		
20	Write a program for binary searching		
21	Write a program for selection sorting		
22	Write a program for bubble sorting		
23	Write a program for insertion sorting		

Program 1: Add two numbers

```
#include <stdio.h>
int main() {
   int a = 5, b = 10, sum;
   sum = a + b;
   printf("Sum = %d\n", sum);
   return 0;
}
```

Output:

```
Sum = 15
```

Program 2: Aggregate marks and percentage of five subjects

```
#include <stdio.h>
int main() {
   int m1=80, m2=76, m3=85, m4=90, m5=70, total;
   float percent;
   total = m1 + m2 + m3 + m4 + m5;
   percent = total / 5.0;
   printf("Total = %d\nPercentage = %.2f%%\n", total, percent);
   return 0;
}
```

Output:

```
Total = 401
Percentage = 80.20%
```

Program 3: Area and perimeter of rectangle

```
#include <stdio.h>
int main() {
    int length = 10, breadth = 5;
    int area = length * breadth;
    int perimeter = 2 * (length + breadth);
    printf("Area = %d\nPerimeter = %d\n", area, perimeter);
    return 0;
}
```

Output:

```
Area = 50
Perimeter = 30
```

Program 4: Simple Interest

```
#include <stdio.h>
int main() {
    float p = 1000, r = 5, t = 2, si;
    si = (p * r * t) / 100;
    printf("Simple Interest = %.2f\n", si);
    return 0;
}
```

Output:

```
Simple Interest = 100.00
```

Program 5: Find larger of two numbers

```
#include <stdio.h>
int main() {
    int a = 20, b = 15;
    if (a > b)
        printf("%d is larger\n", a);
    else
        printf("%d is larger\n", b);
    return 0;
}
```

Output:

```
20 is larger
```

Program 6: Find largest among three numbers

```
#include <stdio.h>
int main() {
   int a = 5, b = 9, c = 3;
   if (a > b && a > c)
      printf("%d is largest\n", a);
   else if (b > c)
      printf("%d is largest\n", b);
   else
      printf("%d is largest\n", c);
   return 0;
}
```

Output:

```
9 is largest
```

Program 7: Check prime number

```
#include <stdio.h>
int main() {
   int n = 7, i, isPrime = 1;
   for(i = 2; i <= n/2; i++) {
      if(n % i == 0) {
         isPrime = 0;
         break;
      }
   }
   if(isPrime && n > 1)
      printf("%d is a prime number\n", n);
   else
      printf("%d is not a prime number\n", n);
   return 0;
}
```

Output:

```
7 is a prime number
```

Program 8: Simple calculator using switch

```
#include <stdio.h>
int main() {
   char op = '+';
   int a = 10, b = 5;
   switch(op) {
      case '+': printf("%d\n", a + b); break;
      case '-': printf("%d\n", a - b); break;
      case '*': printf("%d\n", a * b); break;
      case '/': printf("%d\n", a / b); break;
      default: printf("Invalid operator\n");
   }
   return 0;
}
```

Output:

```
15
```

Program 9: Factorial using loops

```
#include <stdio.h>
int main() {
   int n = 5, i, fact = 1;

   printf("Using for-loop:\n");
   for(i = 1; i <= n; i++) fact *= i;
   printf("Factorial = %d\n", fact);

   fact = 1; i = 1;
   printf("Using while-loop:\n");
   while(i <= n) { fact *= i; i++; }
   printf("Factorial = %d\n", fact);

   fact = 1; i = 1;
   printf("Using do-while loop:\n");
   do { fact *= i; i++; } while(i <= n);
   printf("Factorial = %d\n", fact);
   return 0;
}
```

Output:

```
Using for-loop:
Factorial = 120
Using while-loop:
Factorial = 120
Using do-while loop:
Factorial = 120
```

Program 10: Sum of first n numbers using loops

```
#include <stdio.h>
int main() {
   int n = 10, i, sum = 0;

   printf("Using for-loop:\n");
   for(i = 1; i <= n; i++) sum += i;
   printf("Sum = %d\n", sum);

   i = 1; sum = 0;
   printf("Using while-loop:\n");
   while(i <= n) { sum += i; i++; }
   printf("Sum = %d\n", sum);

   i = 1; sum = 0;
   printf("Using do-while loop:\n");
   do { sum += i; i++; } while(i <= n);
   printf("Sum = %d\n", sum);
   return 0;
}
```

Output:

```
Using for-loop:
Sum = 55
Using while-loop:
Sum = 55
Using do-while loop:
Sum = 55
```

Program 11: Even numbers from 2 to 100

```
#include <stdio.h>
int main() {
   int i;
   for(i = 2; i <= 100; i += 2)
      printf("%d ", i);
   return 0;
}
```

Output:

```
2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50
52 54 56 58 60 62 64 66 68 70 72 74 76 78 80 82 84 86 88 90 92 94 96
98 100
```

Program 12: Print sequence

12.A

```
#include <stdio.h>


int main() {
   int i, j, rows = 5;
for(i = 1; i <= rows; i++) {
       for(j = 1; j <= i; j++) {
           printf("* ");
     }    printf("\n");
   }  return 0;
}
```

Output:

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

12.B

```
#include <stdio.h>
int main() {
   int i, j;
   int rows = 3;

   for(i = 1; i <= rows; i++) {
      for(j = 1; j <= i; j++) {
         printf("%d ", j);
      }
      printf("\n");
   }

   return 0;
}
```

Output:

```
1
1 2
1 2 3
```

Program 13: Factorial using function

```
#include <stdio.h>
int factorial(int n) {
    int fact = 1;
    for(int i = 1; i <= n; i++)
        fact *= i;
    return fact;
}
int main() {
    int n = 5;
    printf("Factorial = %d\n", factorial(n));
    return 0;
}
```

Output:

```
Factorial = 120
```

Program 14: Swap using call by value and reference

```
#include <stdio.h>
void swapValue(int a, int b) {
    int temp = a;
    a = b;
    b = temp;
    printf("Inside swapValue: a = %d, b = %d\n", a, b);
}
void swapReference(int *a, int *b) {
    int temp = *a;
    *a = *b;
    *b = temp;
}
int main() {
    int x = 10, y = 20;
    swapValue(x, y);
    printf("After call by value: x = %d, y = %d\n", x, y);
    swapReference(&x, &y);
    printf("After call by reference: x = %d, y = %d\n", x, y);
    return 0;
}
```

Output:

```
Inside swapValue: a = 20, b = 10
After call by value: x = 10, y = 20
```

After call by reference: x = 20, y = 10

Program 15: Read a string from terminal

```
#include <stdio.h>
int main() {
   char name[^50];
   printf("Enter your name: ");
   gets(name);
   printf("Hello, %s!\n", name);
   return 0;
}
```

Output (example):

```
Enter your name: Alice
Hello, Alice!
```

Program 16: Sum of two matrices

```
#include <stdio.h>
int main() {
   int a[2][2] = {{1, 2}, {3, 4}};
   int b[2][2] = {{5, 6}, {7, 8}};
   int sum[2][2];
   for(int i=0; i<2; i++) {
      for(int j=0; j<2; j++) {
         sum[i][j] = a[i][j] + b[i][j];
         printf("%d ", sum[i][j]);
      }
      printf("\n");
   }
   return 0;
}
```

Output:

```
6 8
10 12
```

Program 17: Multiply two matrices

```
#include <stdio.h>
int main() {
   int a[2][2] = {{1, 2}, {3, 4}};
   int b[2][2] = {{2, 0}, {1, 2}};
   int mul[2][2] = {0};
   for(int i=0; i<2; i++) {
      for(int j=0; j<2; j++) {
         for(int k=0; k<2; k++) {
            mul[i][j] += a[i][k] * b[k][j];
         }
         printf("%d ", mul[i][j]);
      }
      printf("\n");
   }
   return 0;
}
```

Output:

```
4 4
10 8
```

Program 18: Show address using pointers

```
#include <stdio.h>
int main() {
    int a = 10;
    int *p = &a;
    printf("Address of a: %p\n", p);
    printf("Value at address: %d\n", *p);
    return 0;
}
```

Output (address will vary):

```
Address of a: 0x7ffee3b2a6ac
Value at address: 10
```

Program 19: Program for Linear Searching

```
#include <stdio.h>
int main() {
   int arr[] = {2, 4, 6, 8, 10}, n = 5, key = 8, i, found = 0;
   for(i = 0; i < n; i++) {
      if(arr[i] == key) {
         found = 1;
         break;
      }
   }
   if(found)
      printf("Element %d found at position %d\n", key, i+1);
   else
      printf("Element %d not found\n", key);
   return 0;
}
```

Output:

```
Element 8 found at position 4
```

Program 20: Program for Binary Searching

```
#include <stdio.h>
int main() {
   int arr[] = {2, 4, 6, 8, 10}, n = 5, key = 8;
   int low = 0, high = n - 1, mid, found = 0;
   while(low <= high) {
      mid = (low + high) / 2;
      if(arr[mid] == key) {
         found = 1;
         break;
      } else if(arr[mid] < key) {
         low = mid + 1;
      } else {
         high = mid - 1;}
   }
   if(found)
      printf("Element %d found at position %d\n", key, mid+1);
   else
      printf("Element %d not found\n", key);
   return 0;
}
```

Output:

Element 8 found at position 4

Program 21: Program for Selection Sorting

```
#include <stdio.h>
int main() {
   int arr[] = {29, 10, 14, 37, 14}, n = 5, i, j, min, temp;
   for(i = 0; i < n-1; i++) {
      min = i;
      for(j = i+1; j < n; j++) {
         if(arr[j] < arr[min])
            min = j;
      }
      temp = arr[i];
      arr[i] = arr[min];
      arr[min] = temp;
   }
   printf("Sorted array: ");
   for(i = 0; i < n; i++)
      printf("%d ", arr[i]);
   printf("\n");
   return 0;
}
```

Output:

```
Sorted array: 10 14 14 29 37
```

Program 22: Program for Bubble Sorting

```
#include <stdio.h>
int main() {
   int arr[] = {5, 1, 4, 2, 8}, n = 5, i, j, temp;
   for(i = 0; i < n-1; i++) {
      for(j = 0; j < n-i-1; j++) {
         if(arr[j] > arr[j+1]) {
            temp = arr[j];
            arr[j] = arr[j+1];
            arr[j+1] = temp;
         }
      }
   }
   printf("Sorted array: ");
   for(i = 0; i < n; i++)
      printf("%d ", arr[i]);
   printf("\n");
   return 0;
}
```

Output:

```
Sorted array: 1 2 4 5 8
```

Program 23: Program for Insertion Sorting

```
#include <stdio.h>
int main() {
   int arr[] = {12, 11, 13, 5, 6}, n = 5, i, key, j;
   for(i = 1; i < n; i++) {
      key = arr[i];
      j = i - 1;
      while(j >= 0 && arr[j] > key) {
         arr[j+1] = arr[j];
         j = j - 1;
      }
      arr[j+1] = key;
   }
   printf("Sorted array: ");
   for(i = 0; i < n; i++)
      printf("%d ", arr[i]);
   printf("\n");
   return 0;
}
```

Output:

```
Sorted array: 5 6 11 12 13
```