भारतीय सूचना प्रौद्योगिकी संस्थान गुवाहाटी

Indian Institute of Information Technology Guwahati

COMPUTER PROGRAMMING LAB (CS110)
ASSIGNMENTS AND SOLUTIONS–04

[Note: Do not use the `scanf()` function, `switch-case`, and/or `do-while` construct.]

1. Realize the output of the following program:

```c
#include <stdio.h>

int main() {

    int n = 2;

    printf("Line: %d, n = %d\n", __LINE__, n);
    while(printf("Line: %d, n = %d\n", __LINE__, n), n) {
        printf("Line: %d, n = %d\n", __LINE__, n--);
    }
    printf("Line: %d, n = %d\n", __LINE__, n);

    return 0;
}
```

2. Realize the output of the following program:

```c
#include <stdio.h>

int main() {

    int n = 2;

    printf("Line: %d, n = %d\n", __LINE__, n);
    while(n, printf("Line: %d, n = %d\n", __LINE__, n)) { //forever
        printf("Line: %d, n = %d\n", __LINE__, n--);
    }
    printf("Line: %d, n = %d\n", __LINE__, n);

    return 0;
}
```

3. Realize the output of the following program:

```c
#include <stdio.h>

int main() {
    int n = 2;

    printf("Line: %d, n = %d\n", __LINE__, n);
    for (
        printf("Line: %d, n = %d\n", __LINE__, n);
        printf("Line: %d, n = %d\n", __LINE__, n), n;
        printf("Line: %d, n = %d\n", __LINE__, n), n--
    ) {
        printf("Line: %d, n = %d\n", __LINE__, n);
    }
    printf("Line: %d, n = %d\n", __LINE__, n);

    return 0;
}
```

4. Realize the output of the following program:

```c
#include <stdio.h>

int main() {
    int n = 2;

    printf("Line: %d, n = %d\n", __LINE__, n);
    for (
        printf("Line: %d, n = %d\n", __LINE__, n);
        n, printf("Line: %d, n = %d\n", __LINE__, n);
        printf("Line: %d, n = %d\n", __LINE__, n), n--
    ) { //forever
        printf("Line: %d, n = %d\n", __LINE__, n);
    }
    printf("Line: %d, n = %d\n", __LINE__, n);

    return 0;
}
```

5. Write separate programs in C to print the following patterns. Each of them is associated with a control variable `n`. The examples are associated with `n = 4`.

    i. `****`

    ```c
    #include <stdio.h>

    int main() {
        int n = 4;
        for (int i = 1; i <= n; i++) {
            printf("*");
        }
    ```

```
    return 0;
}
```

ii.
#
#
#
#

```
#include <stdio.h>

int main() {
    int n = 4;
    for (int i = 1; i <= n; i++) {
        printf("#\n");
    }
    return 0;
}
```

iii.
$$$$
$$$$
$$$$
$$$$

```
#include <stdio.h>

int main() {
    int n = 4;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n; j++) {
            printf("$");
        }
        printf("\n");
    }
    return 0;
}
```

iv.
?
??
???
????

```
#include <stdio.h>

int main() {
    int n = 4;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= i; j++) {
            printf("?");
        }
```

```
        printf("\n");
    }
    return 0;
}
```

v.
```
   %
  %%
 %%%
%%%%
```

```
#include <stdio.h>

int main() {
    int n = 4;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n - i; j++) {
            printf(" ");
        }
        for (int j = 1; j <= i; j++) {
            printf("%%");
        }
        printf("\n");
    }
    return 0;
}
```

vi.
```
@@@@
@@@
@@
@
```

```
#include <stdio.h>

int main() {
    int n = 4;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n - i + 1; j++) {
            printf("@");
        }
        printf("\n");
    }
    return 0;
}
```

vii.
```
&&&&
 &&&
  &&
   &
```

```c
#include <stdio.h>

int main() {
    int n = 4;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= i - 1; j++) {
            printf(" ");
        }
        for (int j = 1; j <= n - i + 1; j++) {
            printf("&");
        }
        printf("\n");
    }
    return 0;
}
```

viii. 1234
123
12
1

```c
#include <stdio.h>

int main() {
    int n = 4;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n - i + 1; j++) {
            printf("%d", j);
        }
        printf("\n");
    }
    return 0;
}
```

ix. 4321
321
21
1

```c
#include <stdio.h>

int main() {
    int n = 4;
    for (int i = 1; i <= n; i++) {
        int j = 1;
        for (; j <= i - 1; j++) {
            printf(" ");
        }
        for (; j <= n; j++) {
            printf("%d", n - j + 1);
```

```
        }
        printf("\n");
    }
    return 0;
}
```

x. 4
33
222
1111

```
#include <stdio.h>

int main() {
    int n = 4;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= i; j++) {
            printf("%d", n - i + 1);
        }
        printf("\n");
    }
    return 0;
}
```

xi.
```
   0
  01
 012
0123
```

```
#include <stdio.h>

int main() {
    int n = 4;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n - i; j++) {
            printf(" ");
        }
        for (int j = 1; j <= i; j++) {
            printf("%d", j - 1);
        }
        printf("\n");
    }
    return 0;
}
```

xii. For this, consider `n = 5` unlike others.

1
23

456
7890
12345

```c
#include <stdio.h>

int main() {
    int n = 5, digit = 1;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= i; j++) {
            printf("%d", digit);
            digit = (digit + 1) % 10;
        }
        printf("\n");
    }
    return 0;
}
```

xiii. 1
1 2 3
1 2 3 4 5
1 2 3 4 5 6 7
1 2 3 4 5
1 2 3
1

```c
#include <stdio.h>

int main() {
    int n = 4;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= 2 * i - 1; j++) {
            printf("%d ", j);
        }
        printf("\n");
    }
    for (int i = 1; i <= n - 1; i++) {
        for (int j = 1; j <= 2 * (n - i) - 1; j++) {
            printf("%d ", j);
        }
        printf("\n");
    }
    return 0;
}
```

xiv. ****
  *
 *
****

```c
#include <stdio.h>

int main() {
    int n = 4;
    for (int i = 1; i <= n; i++) {
        printf("*");
    }
    printf("\n");
    for (int i = 1; i <= n - 2; i++) {
        for (int j = 1; j <= n - i - 1; j++) {
            printf(" ");
        }
        printf("*\n");
    }
    for (int i = 1; i <= n; i++) {
        printf("*");
    }
    printf("\n");
    return 0;
}
```

xv.
```
****
*  *
*  *
****
```

```c
#include <stdio.h>

int main() {
    int n = 4;
    for (int i = 1; i <= n; i++) {
        printf("*");
    }
    printf("\n");
    for (int i = 1; i <= n - 2; i++) {
        printf("*");
        for (int j = 1; j <= n - 2; j++) {
            printf(" ");
        }
        printf("*\n");
    }
    for (int i = 1; i <= n; i++) {
        printf("*");
    }
    printf("\n");
    return 0;
}
```

xvi.
```
<<<<(      )   >
<<< ((    ))  >>
```

```
<<  (((  ))) >>>
<   ((((()))))>>>>
```

```c
#include <stdio.h>

int main() {
    int n = 4;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n - i + 1; j++)
            printf("<");
        for (int j = 1; j <= i - 1; j++)
            printf(" ");
        for (int j = 1; j <= i; j++)
            printf("(");
        for (int j = 1; j <= n - i; j++)
            printf(" ");
        for (int j = 1; j <= n - i; j++)
            printf(" ");
        for (int j = 1; j <= i; j++)
            printf(")");
        for (int j = 1; j <= n - i; j++)
            printf(" ");
        for (int j = 1; j <= i; j++)
            printf(">");
        printf("\n");
    }
    return 0;
}
```

xvii. 1
010
10101
0101010
10101
010
1

```c
#include <stdio.h>

int main() {
    int n = 4, symbol = 1;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= 2 * i - 1; j++) {
            printf("%d", symbol);
            symbol = (symbol + 1) % 2;
        }
        printf("\n");
    }
    for (int i = 1; i <= n - 1; i++) {
        for (int j = 1; j <= 2 * (n - i) - 1; j++) {
```

```
            printf("%d", symbol);
            symbol = (symbol + 1) % 2;
        }
        printf("\n");
    }
    return 0;
}
```

6. Write separate programs in C to compute the sum of the first n terms of the following series:

   i. $S_1 = 1 + \frac{1}{2} + \frac{1}{3} + \cdots$

```c
#include <stdio.h>

int main() {
    int n = 4;
    double sum = 0.0;
    for (int i = 1; i <= n; i++)
        sum += 1.0 / i;
    printf("%lg", sum);
    return 0;
}
```

   ii. $S_2 = 1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \cdots$

```c
#include <stdio.h>

int main() {
    int n = 4;
    double sum = 0.0, sign = 1.0;
    for (int i = 1; i <= n; i++) {
        sum += sign / i;
        sign *= -1.0;
    }
    printf("%lg", sum);
    return 0;
}
```

   iii. $S_\pi = 4\left(\frac{1}{1} - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots\right)$

```c
#include <stdio.h>

int main() {
    int n = 400;
    double sum = 0.0, sign = 1.0;
    for (int i = 1; i <= n; i++) {
        sum += sign / (2.0 * i - 1.0);
```

```
        sign *= -1.0;
    }
    sum *= 4.0;
    printf("%lg", sum);
    return 0;
}
```

iv. $S_{\log(1+x)} = x - \frac{x^2}{2} + \frac{x^3}{3} - \cdots$

```
#include <stdio.h>

int main() {
    int n = 40;
    double x = 0.3;
    double sum = 0.0, sign = 1.0, x_n = x;
    for (int i = 1; i <= n; i++) {
        sum += (sign * x_n) / i;
        x_n *= x;
        sign *= -1.0;
    }
    printf("%lg", sum);
    return 0;
}
```

v. $S_{e^x} = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$

```
#include <stdio.h>

int main() {
    int n = 40;
    double x = 0.3;
    double sum = 0.0, x_n = 1, n_fac = 1;
    for (int i = 1; i <= n; i++) {
        sum += x_n / n_fac;
        x_n *= x;
        n_fac *= i;
    }
    printf("%lg", sum);
    return 0;
}
```

vi. $S_{\sin(x)} = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \cdots$

```
#include <stdio.h>

int main() {
    int n = 40;
    double x = 0.3;
```

```
    double sum = 0.0, sign = 1.0, x_2n = x, x_2 = x * x, fac = 1.0;
    for (int i = 1; i <= n; i++) {
        sum += (sign * x_2n) / fac;
        sign *= -1.0;
        x_2n *= x_2;
        fac *= (2 * i) * (2 * i + 1);
    }
    printf("%lg", sum);
    return 0;
}
```

vii. $S_{\cos(x)} = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \cdots$

```
#include <stdio.h>

int main() {
    int n = 40;
    double x = 0.3;
    double sum = 0.0, sign = 1.0, x_2n = 1, x_2 = x * x, fac = 1.0;
    for (int i = 1; i <= n; i++) {
        sum += (sign * x_2n) / fac;
        sign *= -1.0;
        x_2n *= x_2;
        fac *= (2 * i - 1) * (2 * i);
    }
    printf("%lg", sum);
    return 0;
}
```

7. A cricket match is going on. The first five overs are done. The runs accumulated in these five overs are stored in five integers variables: `r1`, `r2`, `r3`, `r4`, and `r5`. Write a program in C to print a horizontal bar chart to show runs per over. If the values of these five variables are, respectively, 4, 2, 0, 10, 7, the chart needs to be as follows.

```
Over 1: ####
Over 2: ##
Over 3:
Over 4: ##########
Over 5: #######
```

```
#include <stdio.h>

int main() {
    int r1 = 4, r2 = 2, r3 = 0, r4 = 10, r5 = 7;
    printf("Over 1: ");
    for (int i = 1; i <= r1; i++)
        printf("#");
```

```
    printf("\n");
    printf("Over 2: ");
    for (int i = 1; i <= r2; i++)
        printf("#");
    printf("\n");
    printf("Over 3: ");
    for (int i = 1; i <= r3; i++)
        printf("#");
    printf("\n");
    printf("Over 4: ");
    for (int i = 1; i <= r4; i++)
        printf("#");
    printf("\n");
    printf("Over 5: ");
    for (int i = 1; i <= r5; i++)
        printf("#");
    printf("\n");
    return 0;
}
```

8. Write a program in C to find the least common multiple (LCM) of two numbers.

```
#include <stdio.h>

int main() {
    int a = 6, b = 9;
    int lcm = a > b ? a : b;
    for (; ; lcm++)
        if (lcm % a == 0 && lcm % b == 0)
            break;
    printf("LCM: %d\n", lcm);
    return 0;
}
```

9. Write a program in C to find the greatest common divisor (GCD) of two numbers.

```
#include <stdio.h>

int main() {
    int a = 18, b = 9;
    int gcd = 1;
    for (int i = 1; i <= a && i <= b; i++)
        if (a % i == 0 && b % i == 0)
            gcd = i;
    printf("GCD: %d\n", gcd);
    return 0;
}
```

10. Write a program in C to count the number of digits in a number.

```c
#include <stdio.h>

int main() {
    int n = 10000;
    int number_of_digits = 0;
    for (; n; n /= 10)
        number_of_digits++;
    printf("Number of Digits: %d\n", number_of_digits);
    return 0;
}
```

11. Write a program in C to print all even numbers between $1 - n$, where n is a positive integer.

```c
#include <stdio.h>

int main() {
    int n = 24;
    for (int i = 2; i <= n; i += 2)
        printf("%d\n", i);
    return 0;
}
```

12. Write a program in C to print the multiplication table of any number.

```c
#include <stdio.h>

int main() {
    int n = 5;
    for (int i = 1; i <= 10; i++)
        printf ("%d * %d = %d\n", n, i, n * i);
    return 0;
}
```

13. Write a program in C to print the sum and product of digits of an integer.

```c
#include <stdio.h>

int main() {
    int n = 12345;
    int sum = 0, product = 1;
    for (; n; n /= 10) {
        int digit = n % 10;
        sum += digit;
        product *= digit;
    }
    printf("Sum: %d\n", sum);
    printf("Product: %d\n", product);
    return 0;
}
```

14. Write a program in C to reverse a number.

```c
#include <stdio.h>

int main() {
    int n = 12345;
    int reverse = 0;
    for (; n; n /= 10)
        reverse = reverse * 10 + n % 10;
    printf("Reverse: %d\n", reverse);
    return 0;
}
```

15. Write a program in C to find whether a given positive integer is prime or not.

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

int main() {
    int n = 19;
    int is_prime = 1;
    int sqrt_n = sqrt(n);
    for (int i = 2; i <= sqrt_n; i++) {
        if (n % i == 0) {
            is_prime = 0;
            break;
        }
    }
    printf(is_prime ? "Prime.\n" : "Not Prime.\n");
    return 0;
}
```

16. Write a program in C to print the prime numbers that are less than a given value `n`.

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

int main() {
    int m = 100;
    for (int n = 2; n <= 100; n++) {
        int is_prime = 1;
        int sqrt_n = sqrt(n);
        for (int i = 2; i <= sqrt_n; i++) {
            if (n % i == 0) {
                is_prime = 0;
                break;
            }
        }
        if (is_prime) printf("%d\n", n);
    }
    return 0;
```

```
}
```

17. Write a program in C to find the factorial of a number.

```c
#include <stdio.h>

int main() {
    int n = 6;
    int factorial = 1;
    for (int i = 1; i <= n; i++)
        factorial *= i;
    printf("Factorial: %d\n", factorial);
    return 0;
}
```

18. Write a program in C to check whether a number is a Strong number or not.

```c
#include <stdio.h>

int main() {
    int m = 145;
    int sum = 0, copy = m;
    for (; m; m /= 10) {
        int n = m % 10;
        int factorial = 1;
        for (int i = 1; i <= n; i++)
            factorial *= i;
        sum += factorial;
    }
    printf(copy && copy == sum ? "Strong number.\n" : "Not a strong number.\n");
    return 0;
}
```

19. Write a program in C to print the factors of a given number.

```c
#include <stdio.h>

int main() {
    int n = 12345;
    for (int i = 2; i <= n; i++) {
        if (n % i == 0) {
            printf("%d\n", i);
            n /= i;
            i = 2;
        }
    }
    return 0;
}
```

20. Write a program in C to print the Fibonacci series up to the first n terms.

```c
#include <stdio.h>

int main() {
    int n = 9;
    int p = 0, q = 1;
    for (int i = 1; i <= n; i++) {
        printf("%d : %d\n", i, p);
        int t = p + q;
        p = q;
        q = t;
    }
    return 0;
}
```

21. Write a program in C to find x^n for a given positive real value x and a positive integer n.

```c
#include <stdio.h>

int main() {
    double x = 3.0;
    int n = 3;
    double x_n = 1.0;
    for (int i = 1; i <= n; i++)
        x_n *= x;
    printf("%lg\n", x_n);
    return 0;
}
```

22. Write a program in C to find whether a given number is odd or even. You cannot use the `?:` operator and the `if-else` construct.

```c
#include <stdio.h>

int main() {
    int n = 34;
    double x_n = 1.0;
    for (; n % 2;) {
        printf("%d is odd.\n", n);
        break;
    }
    for (; n % 2 == 0;) {
        printf("%d is even.\n", n);
        break;
    }
    return 0;
}
```

23. Write a program in C to find if a year is a leap year. You cannot use `&&` operaor, `||` operator, `?:` operator, `break`, `continue`, and any `if-else` construct.

```c
#include <stdio.h>

int main() {
    int year = 120;
    for (; year % 400 == 0;) {
        printf("Leap year.\n");
        return 0;
    }
    for (; year % 100 != 0;) {
        for (; year % 4 == 0;) {
            printf("Leap year.\n");
            return 0;
        }
        printf("Not a leap year.\n");
        return 0;
    }
    printf("Not a leap year.\n");
    return 0;
}
```