Purpose: Day 1 ClassWork and Assignment
Date: 19/12/2025
Author: Vinayak Majhi
ID: 58618

Classwork
(NotePad)

Q.1) Check whether given bit position in the number is ON or OFF.

```
                7654    3210
      num --> 10 --> 0000 1010
      pos --> 2
```

If num = 10, pos 2
Output:
2 pos in number 10 is OFF

If num = 10 and pos = 3
Output:
3 pos in number 10 is ON

```c
#include <stdio.h>

int main(){
    int num, pos;
    num = 10, pos = 2;
    int res = (num & 1 << pos);
    printf("%d pos in %d is %s\n", pos, num, (res ? "ON": "OFF"));
}
```

```
assignments/c_cpp_assign/c/Day-1/Code git:main
(base) > gcc 1.c

assignments/c_cpp_assign/c/Day-1/Code git:main
(base) > ./a.out
2 pos in 10 is OFF
```

Q.2) Accept a number from the user, print whether the given number is even or odd.(evenOdd.c)

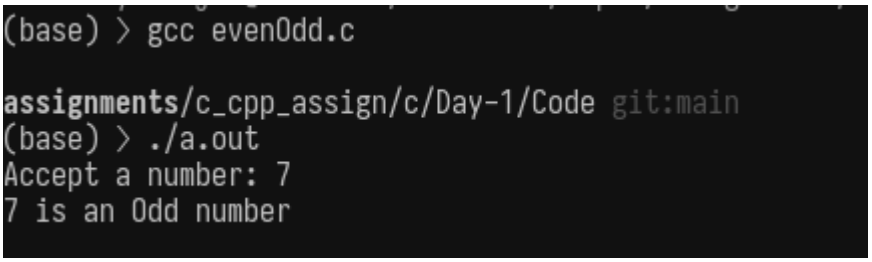
Accept a number: 100

100 is an Even number

Accept a number: 101

101 is an Odd number

```
#include <stdio.h>
int main() {
    int num;

    printf("Accept a number: ");
    scanf("%d", &num);

    if (num % 2 == 0)
        printf("%d is an Even number\n", num);
    else
        printf("%d is an Odd number\n", num);

    return 0;
}
```



Q.2) 2. Adjust your arithmetic operators problem(Day of the week) by adding the conditions on the month, Year. (dayOfWeekV1.c)

Here you can also modify to accept dd,mm,yyyy from the user.

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

```c
int main() {
    int d, m, y;
    int day;

    printf("Enter date (dd mm yyyy): ");
    scanf("%d %d %d", &d, &m, &y);

    if (m < 3) {
        m = m + 12;
        y = y - 1;
    }
    day = (d + (13 * (m + 1)) / 5 + y + (y / 4) - (y / 100) + (y / 400)) % 7;

    printf("Day of the week: ");

    if (day == 0) printf("Saturday\n");
    if (day == 1) printf("Sunday\n");
    if (day == 2) printf("Monday\n");
    if (day == 3) printf("Tuesday\n");
    if (day == 4) printf("Wednesday\n");
    if (day == 5) printf("Thursday\n");
    if (day == 6) printf("Friday\n");

    return 0;
}
```

```
(base) > gcc dayOfWeek.c

assignments/c_cpp_assign/c/Day-1/Code git:main
(base) > ./a.out
Enter date (dd mm yyyy): 18 02 2002
Day of the week: Monday
```

Q.3) 3. Accept a number from the user. Print it with its correct suffix. (numSuffix.c)

Enter the number : 1

Output: 1st

Enter the number : 2
Output: 2nd

Enter the number : 23
Output: 23rd

Enter the number : 111
Output: 111th

```
#include <stdio.h>

int main() {
    int num;
    int lastDigit, lastTwoDigits;

    printf("Enter the number : ");
    scanf("%d", &num);

    lastDigit = num % 10;
    lastTwoDigits = num % 100;

    if (lastTwoDigits >= 11 && lastTwoDigits <= 13)
        printf("Output: %dth\n", num);
    else if (lastDigit == 1)
        printf("Output: %dst\n", num);
    else if (lastDigit == 2)
        printf("Output: %dnd\n", num);
    else if (lastDigit == 3)
        printf("Output: %drd\n", num);
    else
        printf("Output: %dth\n", num);

    return 0;
```

```
assignments/c_cpp_assign/c/Day-1/Code git:main
(base) > ./a.out
Enter the number : 201
Output: 201st
```

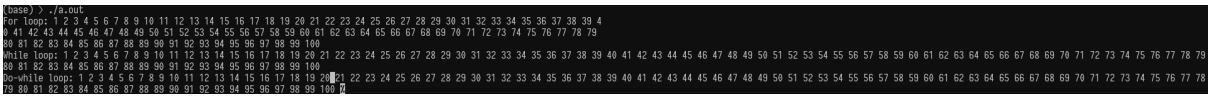
Q.4) Write a program to print 1 .. 100 using all three loops:

for loop: 1 2 3 4 100
while loop: 1 2 3 4 100
do while loop: 1 2 3 4 100

Make sure you use single counter variable(re-use the same variable again in other loop).

```
#include <stdio.h>

int main() {
    int i;

    printf("For loop: ");
    for (i = 1; i <= 100; i++) {
        printf("%d ", i);
    }
    printf("\nWhile loop: ");
    i = 1;
    while (i <= 100) {
        printf("%d ", i);
        i++;
    }

    printf("\nDo-while loop: ");
    i = 1;
    do {
        printf("%d ", i);
```

```
        i++;
    } while (i <= 100);

    return 0;
}
```

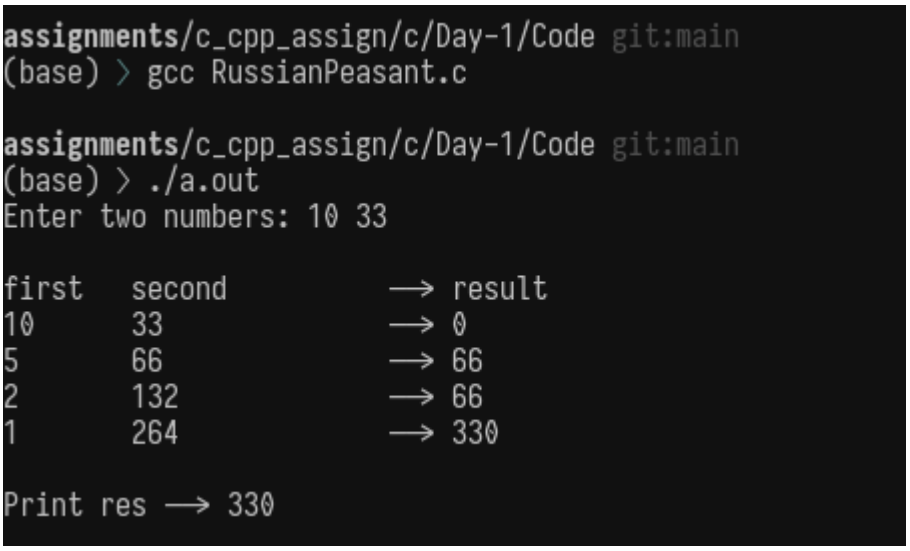


Q.5). Perform multiplication through addition and division using Russian Peasant method. Print along the steps. (RussianPeasant.c)

```
    13 * 24 ===> 312
-------------------------------------------------
    first    second          --> result
    13          24           -->     0 + 24
    6           48           --> 24
    3           96           --> 24 + 96 --> 120
    1           192          --> 120 + 192 --> 312
    Print res --> 312
-------------------------------------------------
```

```
#include <stdio.h>

int main() {
    int first, second;
    int result = 0;

    printf("Enter two numbers: ");
    scanf("%d %d", &first, &second);

    printf("\nfirst\tsecond\t\t--> result\n");

    while (first > 0) {
        printf("%d\t%d\t\t--> ", first, second);
```

```
        if (first % 2 != 0) {
            result = result + second;
            printf("%d\n", result);
        } else {
            printf("%d\n", result);
        }

        first = first / 2;
        second = second * 2;
    }

    printf("\nPrint res --> %d\n", result);

    return 0;
}
```



Q.6. Write a program to add all the numbers which are multiples of 5 & 7 below 1000000(1 million(10 Lakhs)).

```
#include <stdio.h>

int main() {
```

```
    long long sum = 0;
    int i;

    for (i = 35; i < 1000000; i += 35) {
        sum = sum + i;
    }

    printf("Sum of all numbers below 1000000\n");
    printf("that are multiples of 5 and 7 = %lld\n", sum);

    return 0;
}
```

```
assignments/c_cpp_assign/c/Day-1/Code git:main
(base) ❯ ./a.out
Sum of all numbers below 1000000
that are multiples of 5 and 7 = 14285785710
```

Q.7. dayOfWeekV2.c --> After finding the day Of Week using the previous done solution. Extend the same by printing the calendar of any given month and Year.

if for example: Month and Year: 12 2025 --> w --> 1

```
    Su Mo Tu We Th Fr Sa
        1  2  3  4  5  6
    7   8  9 10 11 12 13
    14 15 16 17 18 19 20
    21 22 23 24 25 26 27
    28 29 30
        Just 30 days for every month for the time being.

#include <stdio.h>

int main() {
    int d = 1, m, y;
    int w, i, day;
```

```
    printf("Enter Month and Year: ");
    scanf("%d %d", &m, &y);
    if (m < 3) {
        m = m + 12;
        y = y - 1;
    }

    day = (d + (13 * (m + 1)) / 5 + y + (y / 4) - (y / 100) + (y / 400)) % 7;

    w = (day + 6) % 7;

    printf("\nSu Mo Tu We Th Fr Sa\n");

    for (i = 0; i < w; i++) {
        printf("   ");
    }
    for (i = 1; i <= 30; i++) {
        printf("%2d ", i);

        if ((i + w) % 7 == 0)
            printf("\n");
    }

    printf("\n");

    return 0;
}
```

```
assignments/c_cpp_assign/c/Day-1/Code git:main
(base) > gcc dayofWeekV2.c

assignments/c_cpp_assign/c/Day-1/Code git:main
(base) > ./a.out
Enter Month and Year: 12 2025

Su Mo Tu We Th Fr Sa
    1  2  3  4  5  6
 7  8  9 10 11 12 13
14 15 16 17 18 19 20
21 22 23 24 25 26 27
28 29 30
```

Q.8. Check Whether given number is prime or not?
scenario #1:
Enter a num: 100
100 is NOT a prime
scenario #2:
Enter a num: 101
101 is a prime

```
#include <stdio.h>

int main() {
    int num, i, isPrime = 1;

    printf("Enter a num: ");
    scanf("%d", &num);

    if (num <= 1) {
        isPrime = 0;
    } else {
        for (i = 2; i * i <= num; i++) {
            if (num % i == 0) {
                isPrime = 0;
                break;
            }
        }
```

```
    }

    if (isPrime)
        printf("%d is a prime\n", num);
    else
        printf("%d is NOT a prime\n", num);

    return 0;
}
```

```
assignments/c_cpp_assign/c/Day-1/Code git:main
(base) > ./a.out
Enter a num: 329
329 is NOT a prime

assignments/c_cpp_assign/c/Day-1/Code git:main
(base) > ./a.out
Enter a num: 53
53 is a prime
```

Assignments (Excel Sheet)

Q.1) Create a program to perform simple arithmetic operations and output the results.

```
#include <stdio.h>

int main() {
    int a = 10, b = 5;

    printf("Addition: %d\n", a + b);
    printf("Subtraction: %d\n", a - b);
    printf("Multiplication: %d\n", a * b);
    printf("Division: %d\n", a / b);

    return 0;
```

}

```
assignments/c_cpp_assign/c/Day-1/Code git:main
(base) > ./a.out
Addition: 15
Subtraction: 5
Multiplication: 50
Division: 2

assignments/c_cpp_assign/c/Day-1/Code git:main
(base) >
 TERMINAL  main  $  term:/…/bin/zsh
```

Q.2) Experiment with different data types, observing implicit conversion behavior.

```
#include <stdio.h>

int main() {
    int a = 10;
    float b = 3.5;
    double c = 2.25;

    printf("int + float = %.2f\n", a + b);
    printf("float + double = %.2f\n", b + c);
    printf("int + double = %.2f\n", a + c);

    return 0;
}
```

```
assignments/c_cpp_assign/c/Day-1/Code git:main
(base) > ./a.out
int + float = 13.50
float + double = 5.75
int + double = 12.25
```

Q.3) Construct expressions using various operators and evaluate their results.

```c
#include <stdio.h>

int main() {
    int a = 8, b = 4;

    printf("(a + b) * 2 = %d\n", (a + b) * 2);
    printf("a > b = %d\n", a > b);
    printf("a == b = %d\n", a == b);
    printf("(a > b) && (b > 0) = %d\n", (a > b) && (b > 0));

    return 0;
}
```