NAME: AMAN RAYCHAUDHURI

DIVISION: B2
ROLL NO.: 2401094 (B2-30)
BRANCH: AIDS
FE

Experiment No. 1

Aim: Write a program to print the all even integer between 2-100 which are not divisible by both 8 & 10 also print all the numbers five terms per line

Theory:

1. Initialization:
   Count keeps track of how many numbers are printed on the current line.

2. Iteration:
   The loop starts at 2 and increments by 2 to only check even numbers.

3. Condition:
   The condition !(i % 8 == 0 && i % 10 == 0) ensures the number is not divisible by both 8 and 10 simultaneously.

4. Formatting:
   After every 5 numbers, a new line is printed using printf("\n");.

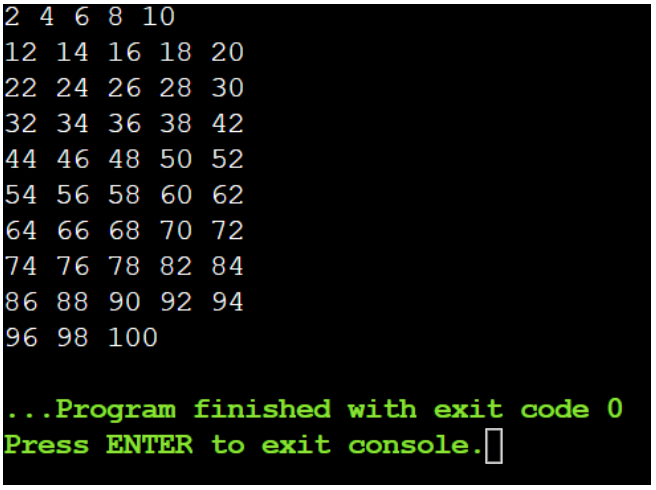
Program:

```
#include <stdio.h>
int main() {
   int count = 0;

   for (int i = 2; i <= 100; i += 2) {
      if (!(i % 8 == 0 && i % 10 == 0)) {
         printf("%d ", i);
         count++;

         if (count % 5 == 0) {
            printf("\n");
         }
      }
   }

   return 0;
}
```

NAME: AMAN RAYCHAUDHURI

DIVISION: B2
ROLL NO.: 2401094 (B2-30)
BRANCH: AIDS
FE

Output:



NAME: AMAN RAYCHAUDHURI

DIVISION: B2
ROLL NO.: 2401094 (B2-30)
BRANCH: AIDS
FE

Experiment No. 2

Aim: Add up and print the numbers divisible by both 8 & 10

Theory:

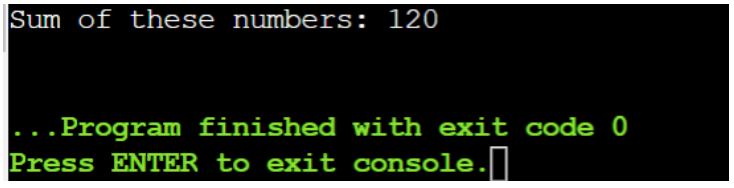
1. Condition:
(i % 8 == 0 && i % 10 == 0): Checks if the number is divisible by both 8 and 10.

2. Output:
Each number is printed as it's found
The sum is accumulated in the sum variable and printed at the end.

Program:

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

   printf("Numbers divisible by both 8 and 10: ");
   for (int i = 2; i <= 100; i++) {
      if (i % 8 == 0 && i % 10 == 0) {
         sum += i;
      }
   }

   printf("\nSum of these numbers: %d\n", sum);

   return 0;
}
```

Output:



NAME: AMAN RAYCHAUDHURI

DIVISION: B2
ROLL NO.: 2401094 (B2-30)
BRANCH: AIDS
FE

Experiment No. 3

Aim: Program to Find the sum of the digits of the number and also find the reverse of the number

Theory:

1. Input:
   The user enters a number, stored in num.

2. Logic:
   A while loop is used to process each digit of the number.
   The remainder of num % 10 gives the last digit.
   Add the digit to sum and build reverse using multiplication and addition.
   Divide num by 10 to remove the processed digit.

3. Output:
   The sum of the digits is stored in sum.
   The reversed number is stored in reverse.

Program:

```
#include <stdio.h>

int main() {
    int num, original, digit, sum = 0, reverse = 0;

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

    original = num;
    while (num > 0) {
        digit = num % 10;
        sum += digit;
        reverse = reverse * 10 + digit;
        num /= 10;
    }


    printf("Sum of the digits: %d\n", sum);
    printf("Reverse of the number: %d\n", reverse);

    return 0;
}
```

NAME: AMAN RAYCHAUDHURI

DIVISION: B2
ROLL NO.: 2401094 (B2-30)
BRANCH: AIDS
FE

Output: