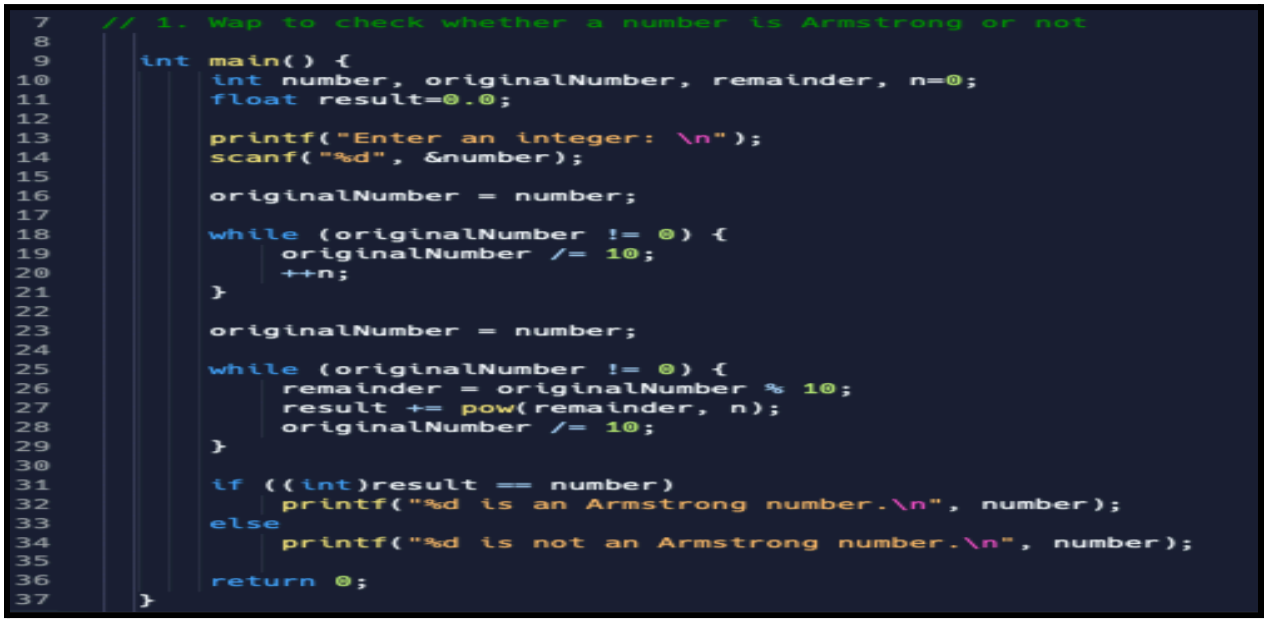
# Assignment-1 (FOCP-1)

### 1. Wap to check whether a number is Armstrong or not

```
int main() {
    int number, originalNumber, remainder, n=0;
    float result=0.0;
    printf("Enter an integer: \n");
    scanf("%d", &number);
    originalNumber = number;
    while (originalNumber != 0) {
        originalNumber /= 10;
        ++n;
    }
    originalNumber = number;
    while (originalNumber != 0) {
        remainder = originalNumber % 10;
        result += pow(remainder, n);
        originalNumber /= 10;
    }
    if ((int)result == number)
        printf("%d is an Armstrong number.\n", number);
    else
        printf("%d is not an Armstrong number.\n", number);
    return 0;
}
```



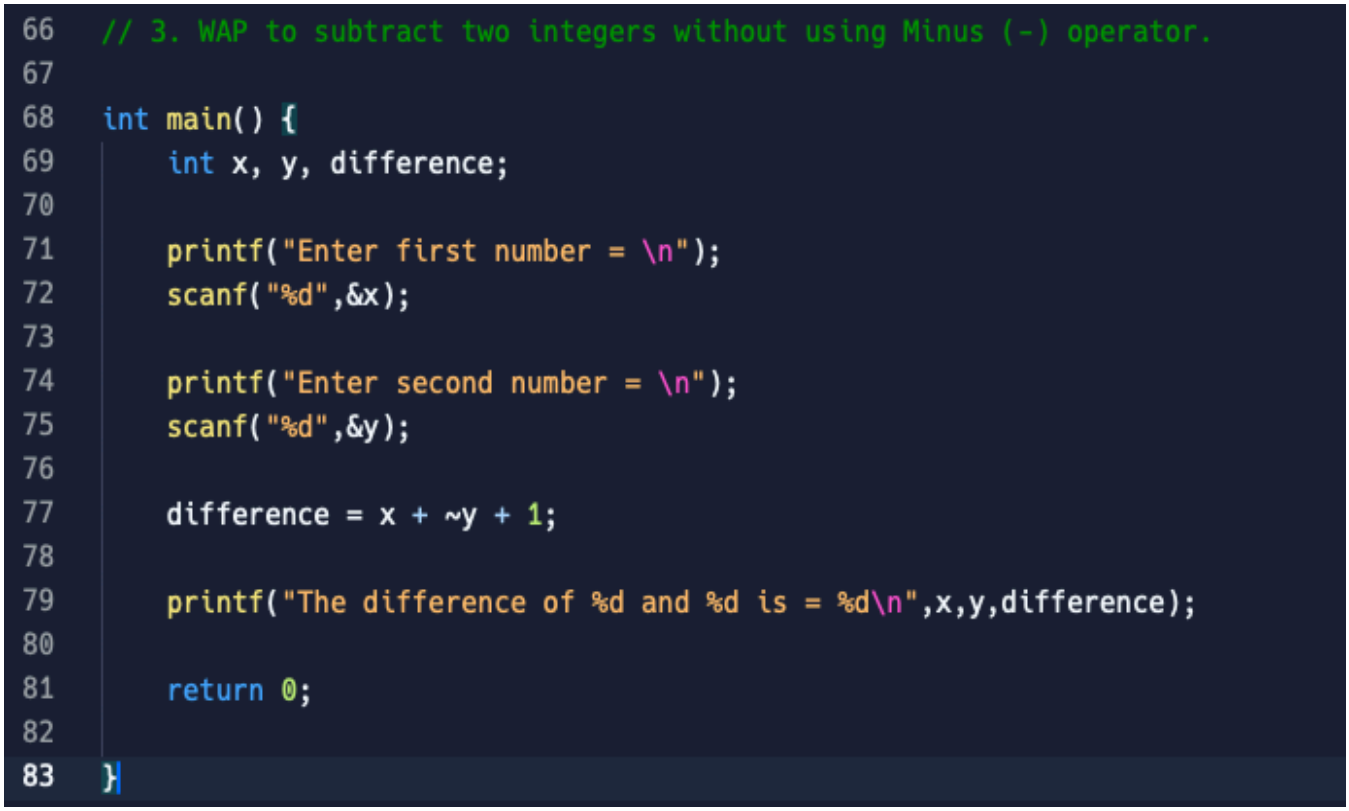
## 2. Wap to read two integers and print their HCF(Highest Common Factor)

```
int main() {
   int i, num1, num2, g, hcf=1;

   printf("Enter first number =  \n");
   scanf("%d",&num1);

   printf("Enter second number = \n");
   scanf("%d",&num2);

   g = (num1<num2)?num1:num2;

   for(i=1; i<=g;i++) {
      if(num1%i==0 && num2%i==0) {
         hcf = i;
      }

   }

   printf("The hcf of %d and %d is =  %d\n",num1,num2,hcf);

   return 0;
}
```

```
40  // 2. WAP to read two integers and print their HCF (Highest Common Factor).
41
42  int main() {
43      int i, num1, num2, g, hcf=1;
44
45      printf("Enter first number =  \n");
46      scanf("%d",&num1);
47
48      printf("Enter second number = \n");
49      scanf("%d",&num2);
50
51      g = (num1<num2)?num1:num2;
52
53      for(i=1; i<=g;i++) {
54          if(num1%i==0 && num2%i==0) {
55              hcf = i;
56          }
57
58      }
59
60      printf("The hcf of %d and %d is =  %d\n",num1,num2,hcf);
61
62      return 0;
63  }
```

### 3. Wap to subtract two integers without using the Minus (-) operator

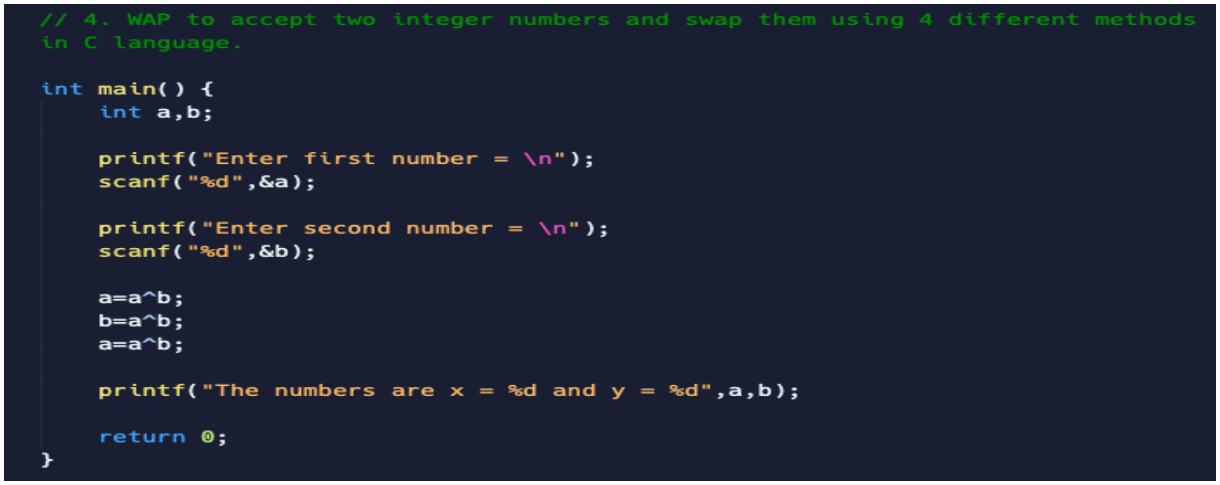
```
int main() {
    int x, y, difference;

    printf("Enter first number = \n");
    scanf("%d",&x);

    printf("Enter second number = \n");
    scanf("%d",&y);

    difference = x + ~y + 1;

    printf("The difference of %d and %d is = %d\n",x,y,difference);

    return 0;

}
```



```
// 3. WAP to subtract two integers without using Minus (-) operator.

int main() {
    int x, y, difference;

    printf("Enter first number = \n");
    scanf("%d",&x);

    printf("Enter second number = \n");
    scanf("%d",&y);

    difference = x + ~y + 1;

    printf("The difference of %d and %d is = %d\n",x,y,difference);

    return 0;

}
```

## 4. Wap to accept two integer numbers swap them using 4 different methods in C language

**(i)**
```
int main() {
    int a,b;

    printf("Enter first number = \n");
    scanf("%d",&a);

    printf("Enter second number = \n");
    scanf("%d",&b);

    a=a^b;
    b=a^b;
    a=a^b;

    printf("The numbers are x = %d and y = %d",a,b);

    return 0;
}
```



```c
// 4. WAP to accept two integer numbers and swap them using 4 different methods
in C language.

int main() {
    int a,b;

    printf("Enter first number = \n");
    scanf("%d",&a);

    printf("Enter second number = \n");
    scanf("%d",&b);

    a=a^b;
    b=a^b;
    a=a^b;

    printf("The numbers are x = %d and y = %d",a,b);

    return 0;
}
```

**(ii)**
```
int main() {
    int a,b,temp;

    printf("Enter first number = \n");
    scanf("%d",&a);

    printf("Enter second number = \n");
    scanf("%d",&b);

    temp=a;
    a=b;
    b=temp;
```

```
    printf("Value of a = %d and b = %d",a,b,temp);

    return 0;
}
```

```
int main() {
    int a,b,temp;

    printf("Enter first number = \n");
    scanf("%d",&a);

    printf("Enter second number = \n");
    scanf("%d",&b);

    temp=a;
    a=b;
    b=temp;
```

**(iii)**
```
int main() {
    int a,b;

    printf("Enter first number = \n");
    scanf("%d",&a);

    printf("Enter second number = \n");
    scanf("%d",&b);

    a=a+b;
    b=a-b;
    a=a-b;

    printf("Value of a = %d and b = %d",a,b);

    return 0;

}
```

```
int main() {
    int a,b;

    printf("Enter first number = \n");
    scanf("%d",&a);

    printf("Enter second number = \n");
    scanf("%d",&b);

    a=a+b;
    b=a-b;
    a=a-b;

    printf("Value of a = %d and b = %d",a,b);

    return 0;
}
```

## 5. Wap to check whether a number is Perfect number or not

```
int main() {
   int number,i,sum=0;

   printf("Enter any number =  \n");
   scanf("%d",&number);

   for(i=1; i<=number/2; i++) {
      if(number%i==0) {
         sum = sum+i;

      }

   }
   if(sum == number && number>0) {
      printf("%d is a Perfect number",number);

   }
   else
   { printf("%d is not a Perfect number",number);
   }
   return 0;
}
```

```
145  // 5. WAP to check whether number is Perfect Number or not
146
147  int main() {
148      int number,i,sum=0;
149
150      printf("Enter any number =  \n");
151      scanf("%d",&number);
152
153      for(i=1; i<=number/2; i++) {
154          if(number%i==0) {
155              sum = sum+i;
156
157          }
158
159      }
160      if(sum == number && number>0) {
161          printf("%d is a Perfect number",number);
162
163      }
164      else
165      { printf("%d is not a Perfect number",number);
166      }
167      return 0;
168  }
```

## 6. WAP to accept a coordinate point in an XY coordinate system and determine in which quadrant the coordinate point lies

```
int main() {
   int x,y;

   printf("Enter the x-coordinate = \n");
   scanf("%d",&x);

   printf("Enter the y-coordinate = \n");
   scanf("%d",&y);

   if(x>0 && y>0){
      printf("The coordinates (%d,%d) lies in the first quadrant\n",x,y);

   }
   else if (x<0 && y>0) {
      printf("The coordinates (%d,%d) lies in the second quadrant \n",x,y);
   }
   else if (x<0 && y<0){
      printf("The coordinates (%d,%d) lies in the third quadrant \n",x,y);
   }
   else if (x>0 && y<0) {
      printf("The coordinates (%d,%d) lies in the fourth quadrant \n",x,y);
      }
   else if (x==0 && y==0) {
      printf("The coordinates (%d,%d) lies at origin\n",x,y);
   }
   return 0;}
```

```
174  int main() {
175      int x,y;
176
177      printf("Enter the x-coordinate = \n");
178      scanf("%d",&x);
179
180      printf("Enter the y-coordinate = \n");
181      scanf("%d",&y);
182
183      if(x>0 && y>0){
184          printf("The coordinates (%d,%d) lies in the first quadrant\n",x,y);
185
186      }
187      else if (x<0 && y>0) {
188          printf("The coordinates (%d,%d) lies in the second quadrant \n",x,y);
189      }
190      else if (x<0 && y<0){
191          printf("The coordinates (%d,%d) lies in the third quadrant \n",x,y);
192      }
193      else if (x>0 && y<0) {
194          printf("The coordinates (%d,%d) lies in the fourth quadrant \n",x,y);
195          }
196      else if (x==0 && y==0) {
197          printf("The coordinates (%d,%d) lies at origin\n",x,y);
198      }
199      return 0;}
```

## 7. WAP for Binary to Decimal conversion & Decimal to Binary for a given number as per user's choice.

```
int binaryToDecimal(long long binary) {
    int decimal = 0, base = 1, remainder;
    while (binary > 0) {
        remainder = binary % 10;
        decimal += remainder * base;
        binary /= 10;
        base *= 2;
    }
    return decimal;
}

long long decimalToBinary(int decimal) {
    long long binary = 0;
    int remainder, i = 1;
    while (decimal > 0) {
        remainder = decimal % 2;
        binary += remainder * i;
        decimal /= 2;
        i *= 10;
    }
    return binary;
}

int main() {
    int choice;
    printf("Choose an option:\n");
    printf("1. Binary to Decimal\n");
    printf("2. Decimal to Binary\n");
    printf("Enter your choice: ");
    scanf("%d", &choice);

    if (choice == 1) {
        long long binary;
        printf("Enter a binary number: ");
        scanf("%lld", &binary);
        int decimal = binaryToDecimal(binary);
        printf("Decimal equivalent: %d\n", decimal);
    } else if (choice == 2) {
        int decimal;
        printf("Enter a decimal number: ");
        scanf("%d", &decimal);
        long long binary = decimalToBinary(decimal);
        printf("Binary equivalent: %lld\n", binary);
    } else {
        printf("Invalid choice!\n");
```

```
    }

    return 0;
}
```

```
int binaryToDecimal(long long binary) {
    int decimal = 0, base = 1, remainder;
    while (binary > 0) {
        remainder = binary % 10;
        decimal += remainder * base;
        binary /= 10;
        base *= 2;
    }
    return decimal;
}

long long decimalToBinary(int decimal) {
    long long binary = 0;
    int remainder, i = 1;
    while (decimal > 0) {
        remainder = decimal % 2;
        binary += remainder * i;
        decimal /= 2;
        i *= 10;
    }
    return binary;
}

int main() {
    int choice;
    printf("Choose an option:\n");
    printf("1. Binary to Decimal\n");
    printf("2. Decimal to Binary\n");
    printf("Enter your choice: ");
    scanf("%d", &choice);

    if (choice == 1) {
        long long binary;
        printf("Enter a binary number: ");
        scanf("%lld", &binary);
        int decimal = binaryToDecimal(binary);
        printf("Decimal equivalent: %d\n", decimal);
    } else if (choice == 2) {
        int decimal;
        printf("Enter a decimal number: ");
        scanf("%d", &decimal);
        long long binary = decimalToBinary(decimal);
        printf("Binary equivalent: %lld\n", binary);
    } else {
        printf("Invalid choice!\n");
    }

    return 0;
}
```

## 8. WAP to print below mentioned pattern:

**1**
**01**
**101**
**0101**
**10101**

```
int main() {
    int rows, i, j;

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

    for (i = 1; i <= rows; i++) {
        for (j = 1; j <= i; j++) {
            if ((i + j) % 2 == 0) {
                printf("1");
            } else {
                printf("0");
            }
        }
        printf("\n");
    }

    return 0;
}
```

```c
// 8. WAP to print below mentioned pattern:
/*1
01
101
0101
10101*/

int main() {
    int rows, i, j;

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

    for (i = 1; i <= rows; i++) {
        for (j = 1; j <= i; j++) {
            if ((i + j) % 2 == 0) {
                printf("1");
            } else {
                printf("0");
            }
        }
        printf("\n");
    }

    return 0;
}
```

## 9. WAP to print following Pyramid:

```
 0          0
 01        01
 010      010
 0101   0101
 0101001010
```

```c
int main() {
    int rows, i, j, k;

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

    for (i = 1; i <= rows; i++) {
        for (j = 1; j <= i; j++) {
            if (j % 2 == 0) {
                printf("1");
            } else {
                printf("0");
            }
        }

        for (k = 1; k <= (2 * (rows - i)); k++) {
            printf(" ");
        }

        for (j = 1; j <= i; j++) {
```

```
            if (j % 2 == 0) {
                printf("1");
            } else {
                printf("0");
            }
        }

        printf("\n");
    }

    return 0;
}
```

```c
int main() {
    int rows, i, j, k;

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

    for (i = 1; i <= rows; i++) {
        for (j = 1; j <= i; j++) {
            if (j % 2 == 0) {
                printf("1");
            } else {
                printf("0");
            }
        }

        for (k = 1; k <= (2 * (rows - i)); k++) {
            printf(" ");
        }

        for (j = 1; j <= i; j++) {
            if (j % 2 == 0) {
                printf("1");
            } else {
                printf("0");
            }
        }

        printf("\n");
    }

    return 0;
}
```

## 10. WAP to print Pascal's Triangle

```
int main() {
    int rows, coef = 1, s, i, j;
    printf("Enter the number of rows: \n");
    scanf("%d", &rows);
    for (i = 0; i < rows; i++) {
        for (s = 1; s <= rows - i; s++)
```

```
      printf("  ");
    for (j = 0; j <= i; j++) {
      if (j == 0 || i == 0)
        coef = 1;
      else
        coef = coef * (i - j + 1) / j;
      printf("%4d", coef);
    }
    printf("\n");
  }
  return 0;
}
```

```
277  // 10. WAP to print Pascal's Triangle
278
279  int main() {
280      int rows, coef = 1, s, i, j;
281      printf("Enter the number of rows: \n");
282      scanf("%d", &rows);
283      for (i = 0; i < rows; i++) {
284          for (s = 1; s <= rows - i; s++)
285              printf("  ");
286          for (j = 0; j <= i; j++) {
287              if (j == 0 || i == 0)
288                  coef = 1;
289              else
290                  coef = coef * (i - j + 1) / j;
291              printf("%4d", coef);
292          }
293          printf("\n");
294      }
295      return 0;
296  }
```