

## Digital assignment 2

Name: Y.Y.N.S.BHARGAV.

Reg.no: 22BRS1248

### First question:

Write a C program to check whether a number is prime, Armstrong, perfect number or not using functions.

Input:

11

Output:

11 is prime number

11 is not an Armstrong number

11 is not a perfect number

### Answer:

```
#include <stdio.h>
#include <math.h>
int is_prime(int n);
int is_armstrong(int n);
int is_perfect(int n);
int main() {
    int n;
    printf("Enter an integer: ");
    scanf("%d", &n);
```

```
if (is_prime(n))
```

```
{
```

```
    printf("%d is a prime number\n", n);
```

```
} else
```

```
{
```

```
    printf("%d is not a prime number\n", n);
```

```
}
```

```
if (is_armstrong(n))
```

```
{
```

```
    printf("%d is an Armstrong number\n", n);
```

```
} else
```

```
{
```

```
    printf("%d is not an Armstrong number\n", n);
```

```
}
```

```
if (is_perfect(n))
```

```
{
```

```
    printf("%d is a perfect number\n", n);
```

```
} else
```

```
{
```

```
printf("%d is not a perfect number\n", n
```

```
}  
    return 0;  
}
```

```
int is_prime(int n) {  
    int i;  
    if (n <= 1) {  
        return 0;  
    }  
    for (i = 2; i <= sqrt(n); i++) {  
        if (n % i == 0) {  
            return 0;  
        }  
    }  
    return 1;  
}
```

```
int is_armstrong(int n) {  
    int sum = 0, temp = n, digits = 0;  
    while (temp > 0) {  
        digits++;  
        temp /= 10;  
    }
```

```
temp = n;
while (temp > 0) {
    int remainder = temp % 10;
    sum += pow(remainder, digits);
    temp /= 10;
}
return (sum == n);
}
```

```
int is_perfect(int n) {
    int i, sum = 0;
    for (i = 1; i < n; i++) {
        if (n % i == 0) {
            sum += i;
        }
    }
    return (sum == n);
}
```