

שיעורי בית יסודות ארגומנטים בשורת הפקודה ותרגילי חזרה – אופיר הופמן י3

```
public static int Factorial(int n)
{
    for (int i = n - 1; i > 1; i--)
    {
        n *= i;
    }
    return n;
}
static void Main(string[] args)
{
    for (int i = 0; i < args.Length; i++)
    {
        if (int.TryParse(args[i]))
            Console.WriteLine(Factorial(int.Parse(s)));
    }
}
```

```
public static bool IsPerfect(int[] arr)
{
    int index = 0;
    int cnt = 0;

    bool foundZero = false;

    for (int i = 0; i < arr.Length && !foundZero; i++)
    {
        if (arr[index] == 0)
            foundZero = true;

        index = arr[index];
        cnt++;
    }

    return cnt == arr.Length && foundZero;
}
```

```
public bool IsRed()
{
    return this.red > 0 && this.green == 0 && this.blue == 0;
}
```

```
public bool IsBalanced()
{
    int red = 0;
    int green = 0;
    int blue = 0;
}
```

```

for (int i = 0; i < this.arr.Length; i++)
{
    if (arr[i].IsRed())
        red++;
    else if (arr[i].IsGreen())
        green++;
    else
        (arr[i].IsBlue())
            blue++;
}

return red == green && green == blue;
}

public static bool SimetricSum(int[] arr, int i)
{
    if (i == arr.Length/2)
        return true;

    return arr[i] + arr[arr.Length-1-i] == arr[i + 1] + arr[arr.Length-2-i] && SimetricSum(arr, i+1);
}

public static bool SimetricSum(int[] arr)
{
    return SimetricSum(arr, 0);
}

```

4. ב. מטרת התוכנית היא להדפיס את כל המספרים הבינאריים שניתן לכתוב במספר ביטים שקיבלה התוכנית בפרמטר m. (טבלת מעקב מצורפת בטימס).