



Case Study #1

OPERATING SYSTEMS [Section 53]



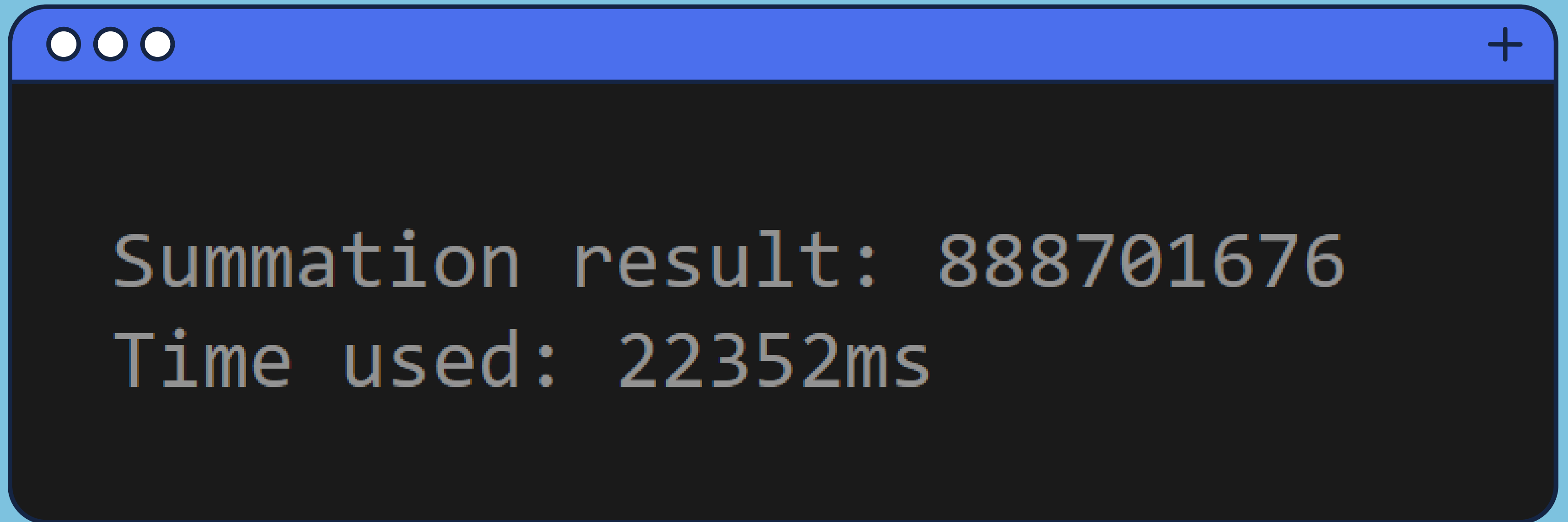


สมาชิกในกลุ่ม

- 64015019 นายจารุพัฒน์ เคนพรม
- 64015051 นายเตชินท์ ไม้ทอง
- 64015102 นายพิสิฐพงศ์ พิสิฐแก้วเพชร
- 64015112 นายภูมิพัฒน์ ลาลุน
- 64015163 นายอภิสิทธิ์ชัย ทองโต
- 64015172 นายเอกรินทร์ องอาจ



Version Original



A terminal window with a blue title bar and a black body. The title bar contains three white window control buttons on the left and a white plus sign on the right. The terminal body displays two lines of text in a light blue monospace font.

```
Summation result: 888701676  
Time used: 22352ms
```

```

static void sum()
{
    for (int G_index = 0; G_index < 1000000000; G_index++)
    {
        if (Data_Global[G_index] % 2 == 0)
        {
            Sum_Global -= Data_Global[G_index];
        }
        else if (Data_Global[G_index] % 3 == 0)
        {
            Sum_Global += (Data_Global[G_index] * 2);
        }
        else if (Data_Global[G_index] % 5 == 0)
        {
            Sum_Global += (Data_Global[G_index] / 2);
        }
        else if (Data_Global[G_index] % 7 == 0)
        {
            Sum_Global += (Data_Global[G_index] / 3);
        }
        Data_Global[G_index] = 0;
    }
}

```

ส่วนที่คิดว่าเป็นปัญหา

1. มีการเรียกใช้งานฟังก์ชันบ่อยเกินไป



```

sw.Start();
for (i = 0; i < 1000000000; i++)
{
    sum();
}
sw.Stop();

```

Version 1

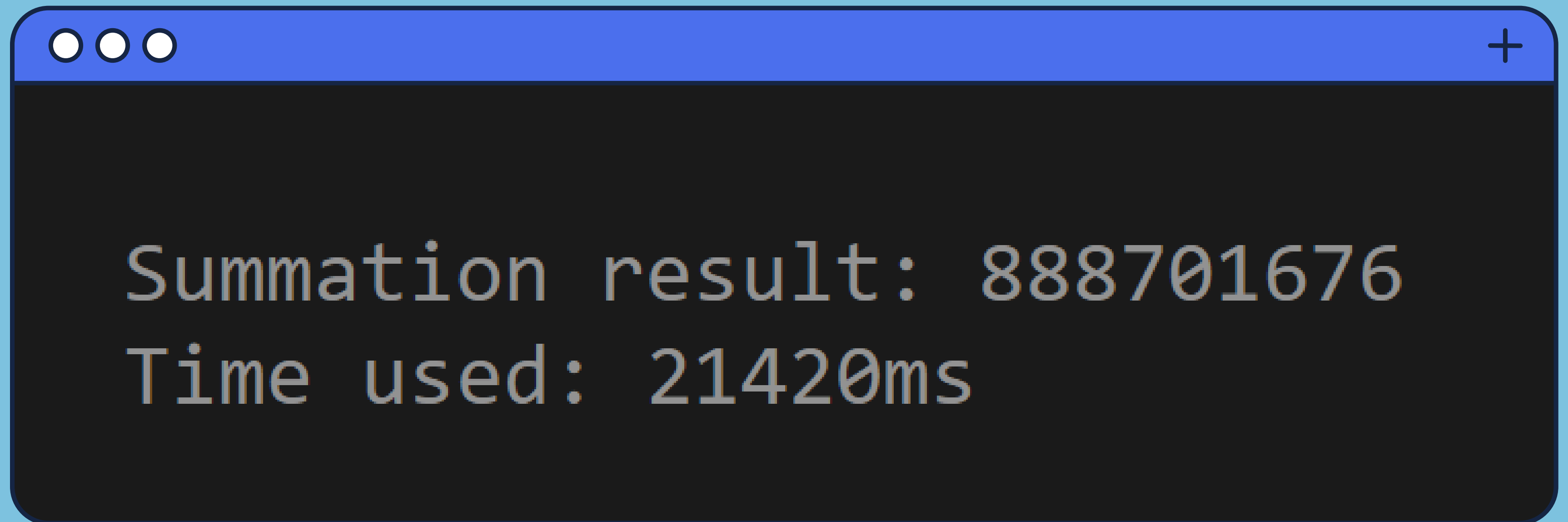
```
1 reference
static void sum()
{
    for (int G_index = 0; G_index < 1000000000; G_index++)
    {
        if (Data_Global[G_index] % 2 == 0)
        {
            Sum_Global -= Data_Global[G_index];
        }
        else if (Data_Global[G_index] % 3 == 0)
        {
            Sum_Global += (Data_Global[G_index] * 2);
        }
        else if (Data_Global[G_index] % 5 == 0)
        {
            Sum_Global += (Data_Global[G_index] / 2);
        }
        else if (Data_Global[G_index] % 7 == 0)
        {
            Sum_Global += (Data_Global[G_index] / 3);
        }
        Data_Global[G_index] = 0;
    }
}
```

แก้ไขโดย

ย้าย Loop for เข้าไปใน Function sum()
เพราะอยากให้เรียกใช้ฟังก์ชันเพียงครั้งเดียว

```
/* Start */
Console.WriteLine("\n\nWorking...");
sw.Start();
sum();
sw.Stop();
Console.WriteLine("Done.");
```

Version 1 (Loop for)



Version Original Time Used: 22352ms

```

static void sum()
{
    for (int G_index = 0; G_index < 1000000000; G_index++)
    {
        if (Data_Global[G_index] % 2 == 0)
        {
            Sum_Global -= Data_Global[G_index];
        }
        else if (Data_Global[G_index] % 3 == 0)
        {
            Sum_Global += (Data_Global[G_index] * 2);
        }
        else if (Data_Global[G_index] % 5 == 0)
        {
            Sum_Global += (Data_Global[G_index] / 2);
        }
        else if (Data_Global[G_index] % 7 == 0)
        {
            Sum_Global += (Data_Global[G_index] / 3);
        }
        Data_Global[G_index] = 0;
    }
}

```

ส่วนที่คิดว่าเป็นปัญหา

1. ใน Function sum() ใช้ Loop for วนลูปหลายรอบเกินไปในการกำหนดตำแหน่งของ Array และ ไม่มีการแบ่งการทำงาน

```

/* Start */
Console.WriteLine("\n\nWorking...");
sw.Start();
sum();
sw.Stop();
Console.WriteLine("Done.");

```

Version 2

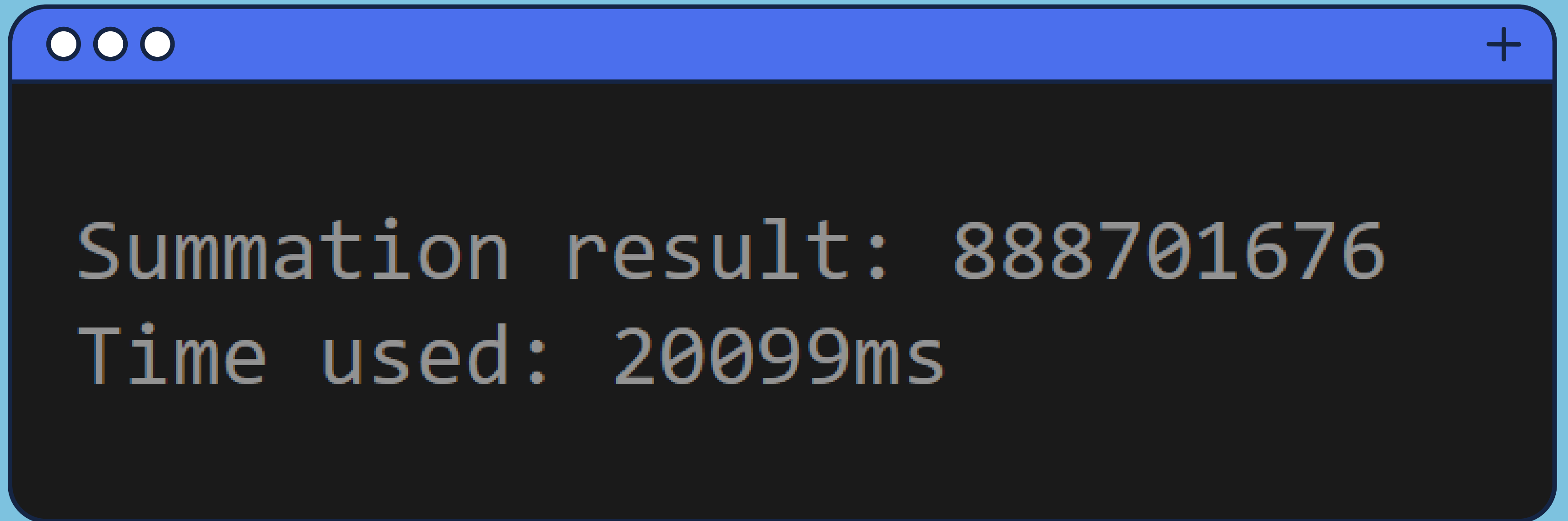
```
1 reference
static void sum()
{
    int G_index = 0;
    for (int i = 0; i < 1000000000; i++)
    {
        if (Data_Global[G_index] % 2 == 0)
        {
            Sum_Global -= Data_Global[G_index];
        }
        else if (Data_Global[G_index] % 3 == 0)
        {
            Sum_Global += (Data_Global[G_index] * 2);
        }
        else if (Data_Global[G_index] % 5 == 0)
        {
            Sum_Global += (Data_Global[G_index] / 2);
        }
        else if (Data_Global[G_index] % 7 == 0)
        {
            Sum_Global += (Data_Global[G_index] / 3);
        }
        Data_Global[G_index] = 0;
        G_index += 1;
    }
}
```

แก้ไขโดย

เพิ่ม Thread เป็น Single Thread
เพื่อเริ่มแบ่งการทำงานเป็น MultiThread

```
/* Start */
Console.Write("\n\nWorking...");
sw.Start();
th1.Start();
th1.Join();
sw.Stop();
Console.WriteLine("Done.");
```


Version 2 (Single Thread)

A terminal window with a blue title bar and three white window control buttons on the left. The main area is black with yellow text. It displays the summation result and the time used for Version 2.

```
Summation result: 888701676  
Time used: 20099ms
```

Version 1 Time Used: 21420ms

ส่วนที่คิดว่าเป็นปัญหา

1. การใช้ Single Thread ยังไม่ช่วยให้เร็วขึ้น

```
static void sum()
{
    int G_index = 0;
    for (int i = 0; i < 1000000000; i++)
    {
        if (Data_Global[G_index] % 2 == 0)
        {
            Sum_Global -= Data_Global[G_index];
        }
        else if (Data_Global[G_index] % 3 == 0)
        {
            Sum_Global += (Data_Global[G_index] * 2);
        }
        else if (Data_Global[G_index] % 5 == 0)
        {
            Sum_Global += (Data_Global[G_index] / 2);
        }
        else if (Data_Global[G_index] % 7 == 0)
        {
            Sum_Global += (Data_Global[G_index] / 3);
        }
        Data_Global[G_index] = 0;
        G_index += 1;
    }
}
```

```
Console.WriteLine("\n\nWorking...");
sw.Start();
th1.Start();
th1.Join();
sw.Stop();
Console.WriteLine("Done.");
```

Version 3

```
static void sum1()
{
    int G_index1 = 0;
    for (int i = 0; i < 500000000; i++)
    {
        if (Data_Global[G_index1] % 2 == 0)
        {
            Sum_Global1 -= Data_Global[G_index1];
        }
        else if (Data_Global[G_index1] % 3 == 0)
        {
            Sum_Global1 += (Data_Global[G_index1] * 2);
        }
        else if (Data_Global[G_index1] % 5 == 0)
        {
            Sum_Global1 += (Data_Global[G_index1] / 2);
        }
        else if (Data_Global[G_index1] % 7 == 0)
        {
            Sum_Global1 += (Data_Global[G_index1] / 3);
        }
        Data_Global[G_index1] = 0;
        G_index1+=2;
    }
}

1 reference
static void sum2()
{
    int G_index2 = 1;
    for (int i = 0; i < 500000000; i++)
    {
        if (Data_Global[G_index2] % 2 == 0)
        {
            Sum_Global2 -= Data_Global[G_index2];
        }
        else if (Data_Global[G_index2] % 3 == 0)
        {
            Sum_Global2 += (Data_Global[G_index2] * 2);
        }
        else if (Data_Global[G_index2] % 5 == 0)
        {
            Sum_Global2 += (Data_Global[G_index2] / 2);
        }
        else if (Data_Global[G_index2] % 7 == 0)
        {
            Sum_Global2 += (Data_Global[G_index2] / 3);
        }
        Data_Global[G_index2] = 0;
        G_index2+=2;
    }
}
```

แก้ไขโดย

เพิ่ม Thread เป็น MultiThread[2]

```
/* Start */
Console.WriteLine("\n\nWorking...");
sw.Start();
th1.Start();
th2.Start();
th1.Join();
th2.Join();
sw.Stop();
Console.WriteLine("Done.");
```

วิธีทำ Multi Threads

- เพิ่มตัว Sum_Global ให้เท่ากับจำนวน thread ที่ใช้งาน
- แบ่ง Data_Global ให้มีขนาดเท่ากับจำนวน thread (1พันล้าน/thread)
- แต่ละ thread จะบวกค่าในส่วนของตัวเอง และเก็บไว้ที่ Sum_Global ของตัวเอง

```
16 namespace Problem01
17 {
18     0 references
19     class Program
20     {
21         74 references
22         static byte[] Data_Global = new byte[1000000000]; // 1 GB of data
23         5 references | 5 references
24         static long Sum_Global1 = 0, Sum_Global2 = 0; //เพิ่มตัว Sum_Global ให้เท่ากับจำนวนของ Thread ที่ใช้งาน
```

วิธีทำ Multi Threads

- เพิ่มตัว Sum_Global ให้เท่ากับจำนวน thread ที่ใช้งาน
- แบ่ง Data_Global ให้มีขนาดเท่ากับจำนวน thread (1พันล้าน/thread)
- แต่ละ thread จะบวกค่าในส่วนของตัวเอง และเก็บไว้ที่ Sum_Global ของตัวเอง

```
1 reference
static void sum2()
{
    int G_index2 = 1;
    for (int i = 0; i < 500000000; i++) //แบ่ง Data_Global ให้มีขนาดเท่ากับจำนวน thread
    {
        if (Data_Global[G_index2] % 2 == 0)
        {
            Sum_Global2 -= Data_Global[G_index2];
        }
    }
}
```

วิธีทำ Multi Threads

- เพิ่มตัว Sum_Global ให้เท่ากับจำนวน thread ที่ใช้งาน
- แบ่ง Data_Global ให้มีขนาดเท่ากับจำนวน thread (1พันล้าน/thread)
- แต่ละ thread จะบวกค่าในส่วนของตัวเอง และเก็บไว้ที่ Sum_Global ของตัวเอง

```
73     else if (Data_Global[G_index2] % 3 == 0)
74     {
75         Sum_Global2 += (Data_Global[G_index2] * 2); //แต่ละ thread จะบวกค่าในส่วนของตัวเอง
76     }
77     else if (Data_Global[G_index2] % 5 == 0)
78     {
79         Sum_Global2 += (Data_Global[G_index2] / 2); //แต่ละ thread จะบวกค่าในส่วนของตัวเอง
80     }
81     else if (Data_Global[G_index2] % 7 == 0)
82     {
83         Sum_Global2 += (Data_Global[G_index2] / 3); //แต่ละ thread จะบวกค่าในส่วนของตัวเอง
84     }
85     Data_Global[G_index2] = 0;
86     G_index2+=2;
87 }
```

●●● Program.cs

```
1  using System;
2  using System.Diagnostics;
3  using System.IO;
4  using System.Runtime.Serialization;
5  using System.Runtime.Serialization.Formatters.Binary;
6  using System.Threading;
7
8  namespace Problem01
9  {
10     class Program
11     {
12         static byte[] Data_Global = new byte[1000000000];
13         static long Sum_Global1 = 0 , Sum_Global2 = 0;
```

```
15  static int ReadData()
16      {
17          int returnData = 0;
18          FileStream fs = new FileStream("Problem01.dat", FileMode.Open);
19          BinaryFormatter bf = new BinaryFormatter();
20
21          try
22          {
23              Data_Global = (byte[])bf.Deserialize(fs);
24          }
25          catch (SerializationException se)
26          {
27              Console.WriteLine("Read Failed:" + se.Message);
28              returnData = 1;
29          }
30          finally
31          {
32              fs.Close();
33          }
34
35          return returnData;
36      }
```



```
37 static void sum1()
38     {
39         int G_index1 = 0;
40         for (int i = 0; i < 500000000; i++)
41         {
42             if (Data_Global[G_index1] % 2 == 0)
43             {
44                 Sum_Global1 -= Data_Global[G_index1];
45             }
46             else if (Data_Global[G_index1] % 3 == 0)
47             {
48                 Sum_Global1 += (Data_Global[G_index1] * 2);
49             }
50             else if (Data_Global[G_index1] % 5 == 0)
51             {
52                 Sum_Global1 += (Data_Global[G_index1] / 2);
53             }
54             else if (Data_Global[G_index1] % 7 == 0)
55             {
56                 Sum_Global1 += (Data_Global[G_index1] / 3);
57             }
58             Data_Global[G_index1] = 0;
59             G_index1+=2;
60         }
61     }
```

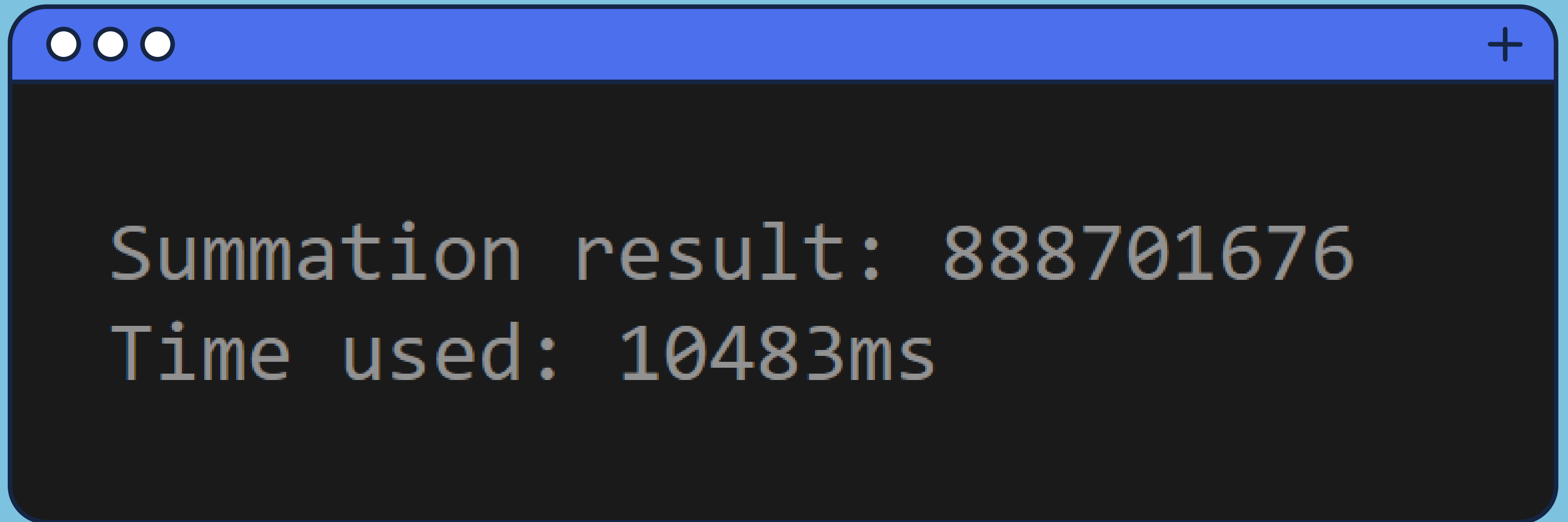
```
63 static void sum2()
64     {
65         int G_index2 = 1;
66         for (int i = 0; i < 500000000; i++)
67         {
68             if (Data_Global[G_index2] % 2 == 0)
69             {
70                 Sum_Global2 -= Data_Global[G_index2];
71             }
72             else if (Data_Global[G_index2] % 3 == 0)
73             {
74                 Sum_Global2 += (Data_Global[G_index2] * 2);
75             }
76             else if (Data_Global[G_index2] % 5 == 0)
77             {
78                 Sum_Global2 += (Data_Global[G_index2] / 2);
79             }
80             else if (Data_Global[G_index2] % 7 == 0)
81             {
82                 Sum_Global2 += (Data_Global[G_index2] / 3);
83             }
84             Data_Global[G_index2] = 0;
85             G_index2+=2;
86         }
87     }
```

```
89     static void Main(string[] args)
90     {
91         Stopwatch sw = new Stopwatch();
92         Thread th1 = new Thread(sum1);
93         Thread th2 = new Thread(sum2);
94         /* Read data from file */
95         Console.Write("Data read...");
96         int y = ReadData();
97         if (y == 0)
98         {
99             Console.WriteLine("Complete.");
100        }
101        else
102        {
103            Console.WriteLine("Read Failed!");
104        }
```

●●● Program.cs

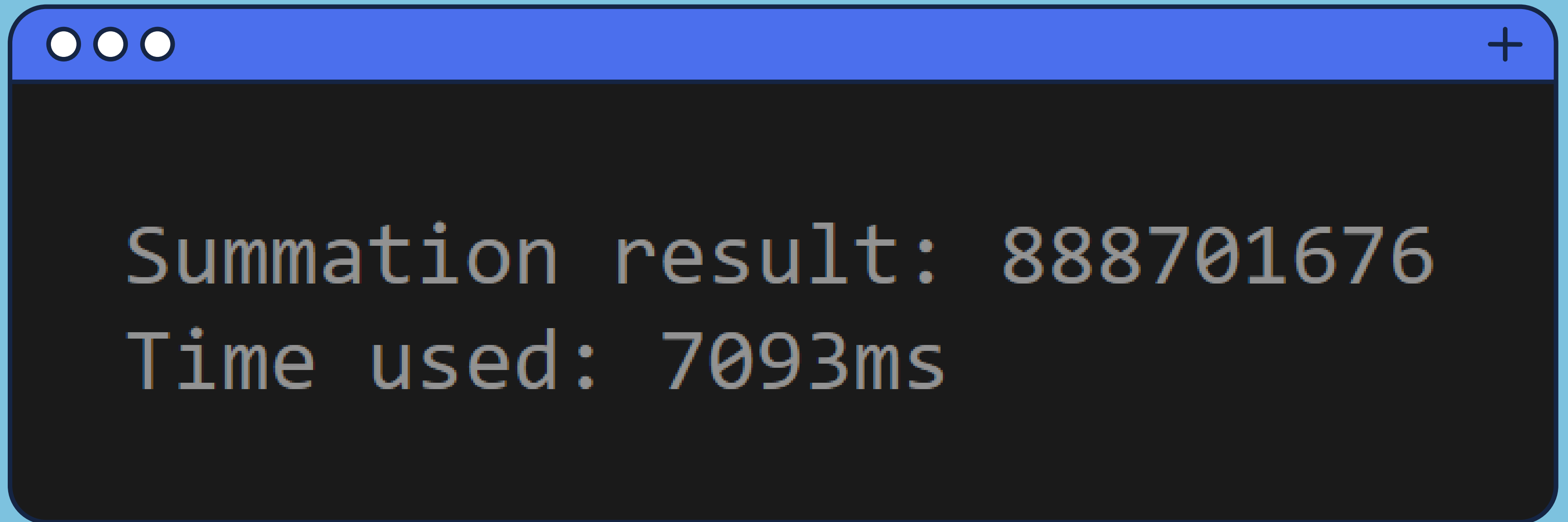
```
106  /* Start */
107      Console.Write("\n\nWorking...");
108      sw.Start();
109      th1.Start();
110      th2.Start();
111      th1.Join();
112      th2.Join();
113      sw.Stop();
114      Console.WriteLine("Done.");
115
116      /* Result */
117      Console.WriteLine("Summation result: {0}", Sum_Global1 + Sum_Global2);
118      Console.WriteLine("Time used: " + sw.ElapsedMilliseconds.ToString() + "ms");
119  }
120  }
121  }
```

Version 3 (MultiThread[2])



Version 2 Time Used: 20099ms

Version 4 (Mutithread[8])



```
Summation result: 888701676  
Time used: 7093ms
```

A terminal window with a blue title bar and a black body. The title bar contains three white window control buttons on the left and a white plus sign on the right. The terminal body displays two lines of text in a light blue monospace font: "Summation result: 888701676" and "Time used: 7093ms".

Version 3 Time Used: 10483ms

สรุปการทดสอบ

- ยิ่งเพิ่ม threads เวลาที่ใช้ยิ่งลดลง
- **Speed** แล่มาด้วย **Space**

Result :

1

Thread

```
Summation result: 888701676  
Time used: 20099ms
```

2

Threads

```
Summation result: 888701676  
Time used: 10483ms
```

4

Threads

```
Summation result: 888701676  
Time used: 7934ms
```

8

Threads

```
Summation result: 888701676  
Time used: 7093ms
```



ขอบคุณครับ

Have a good day.