Day 2 – Task 1

Cataloguing numbers

We repeat the same algorithm for every number from the input. First we split the number from the input into 5 numbers: day, month, year, author and control number. We have to determine the exact year before checking if it's leap or not. If the three-digit number for year is smaller than 600, it is the year from 2000 to 2599 so we add 2000, otherwise we add 1000. Next we check if the month is between 1 and 12 and if the day is between 1 and the number of days in that month. Since author number can be only 1, 6 or 9 we check if it is one of those three. Finally, we split the number into digits and sum square of the first 8 digits. We compare remainder after divison by 7 of the sum of squares to the nineth digit. If those match we write 1. If any of the checks failed we write 0.

Day 2 – Task 1

Example solution in Pascal:

```pascal
var
  n, day, month, year, author, control, ndays, data, sum: Longint;
  leap : Boolean;
  valid : Boolean;
  i, count : Integer;
begin
  for count := 1 to 3 do begin
    ReadLn (n);

    day     := (n div 10000000) mod 100;
    month   := (n div   100000) mod 100;
    year    := (n div      100) mod 1000;
    author  := (n div       10) mod 10;
    control := (n div        1) mod 10;
    data    := (n div       10);

    if (year >= 600) then
      year := year + 1000
    else
      year := year + 2000;

    if (year mod 4 = 0) then begin
      if (year mod 100 = 0) then
        if (year mod 400 = 0) then
          leap := true
        else
          leap := false
      else
        leap := true;
    end
    else
      leap := false;

    if (month in [1, 3, 5, 7, 8, 10, 12]) then
      ndays := 31
    else
      if (month = 2) then
        if (leap) then
          ndays := 29
        else
          ndays := 28
      else
        ndays := 30;
        sum := 0;
        for i := 1 to 8 do begin
          sum := sum + Sqr(data mod 10);
          data := data div 10;
        end;

        valid := true;
        valid := valid and (month >= 1) and (month <= 12);
        valid := valid and (day >= 1) and (day <= ndays);
        valid := valid and (author in [1, 6, 9]);
        valid := valid and (sum mod 7 = control);
        if (valid) then
          WriteLn (1)
        else
          WriteLn (0);
  end;
end.
```

Day 2 – Task 1

Example solution in C#:

```csharp
using System;
using System.IO;

class Program
{
    static void Main(string[] args)
    {
        int num, d, m, y, type, checksum, sum, temp;
        bool ok, leapYear;

        for (int i = 0; i < 3; i++)
        {
            num = Convert.ToInt32(Console.ReadLine());
            temp = num / 10;
            checksum = num % 10;
            num /= 10;
            type = num % 10;
            num /= 10;
            y = num % 1000;
            num /= 1000;
            m = num % 100;
            num /= 100;
            d = num;
            ok = (0 < d) && (0 < m) && (m <= 12) && ((type == 1) || (type == 6) || (type == 9));
            if (ok)
            {
                if (y >= 600)
                    y += 1000;
                else
                    y += 2000;
                leapYear = (y % 4 == 0) && ((y % 100 != 0) || (y % 400 == 0));
                if ((m == 1) || (m == 3) || (m == 5) ||
                (m == 7) || (m == 8) || (m == 10) || (m == 12))
                    ok &= d <= 31;
                else if ((m == 4) || (m == 6) || (m == 9) || (m == 11))
                    ok &= d <= 30;
                else
                    ok &= (d <= 28) || ((d == 29) && leapYear);
            }
            if (ok)
            {
                sum = 0;
                while (temp > 0)
                {
                    sum += (temp % 10) * (temp % 10);
                    temp /= 10;
                }
                ok = checksum == sum % 7;
            }
            if (ok)
                Console.WriteLine(1);
            else
                Console.WriteLine(0);
        }
    }
}
```