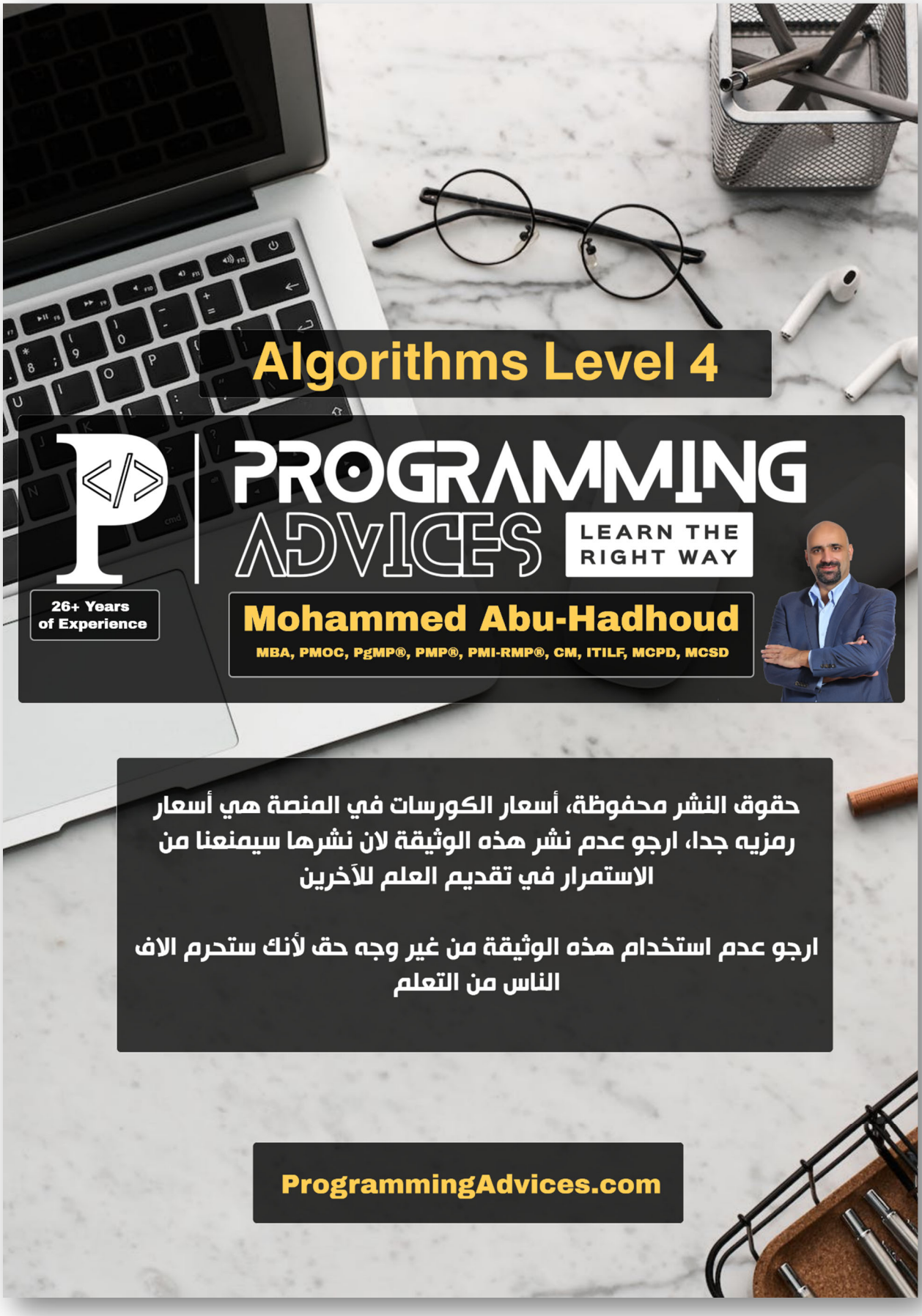
Algorithms Level 4
PROGRAMMING ADVICES
LEARN THE RIGHT WAY
26+ Years of Experience
Mohammed Abu-Hadhoud
MBA, PMOC, PgMP®, PMP®, PMI-RMP®, CM, ITILF, MCPD, MCSD
حقوق النشر محفوظة، أسعار الكورسات في المنصة هي أسعار رمزيه جدا، ارجو عدم نشر هذه الوثيقة لان نشرها سيمنعنا من الاستمرار في تقديم العلم للآخرين
ارجو عدم استخدام هذه الوثيقة من غير وجه حق لأنك ستحرم الاف الناس من التعلم
ProgrammingAdvices.com

```cpp
#include <iostream>
using namespace std;

struct stDate
{
    short Year;
    short Month;
    short Day;

};

bool isLeapYear(short Year)
{
   return (Year % 4 == 0 && Year % 100 != 0) || (Year % 400 == 0);
}

bool IsDate1BeforeDate2(stDate Date1, stDate Date2)
{
    return  (Date1.Year < Date2.Year) ? true : ((Date1.Year ==
Date2.Year) ? (Date1.Month < Date2.Month ? true : (Date1.Month ==
Date2.Month ? Date1.Day < Date2.Day : false)) : false);
}

short NumberOfDaysInAMonth(short Month, short Year)
{

    if (Month < 1 || Month>12)
        return  0;

    int days[12] = { 31,28,31,30,31,30,31,31,30,31,30,31 };
    return (Month == 2) ? (isLeapYear(Year) ? 29 : 28) :
days[Month - 1];

}

bool IsLastDayInMonth(stDate Date)
{

    return (Date.Day == NumberOfDaysInAMonth(Date.Month,
Date.Year));

}

bool IsLastMonthInYear(short Month)
{
    return (Month == 12);
}
```

```
stDate IncreaseDateByOneDay(stDate Date)
{
    if (IsLastDayInMonth(Date))
    {
        if (IsLastMonthInYear(Date.Month))
        {
            Date.Month = 1;
            Date.Day = 1;
            Date.Year++;
        }
        else
        {
            Date.Day = 1;
            Date.Month++;
        }
    }
    else
    {
        Date.Day++;
    }

    return Date;
}

void  SwapDates(stDate& Date1, stDate& Date2)
{

    stDate TempDate;

    TempDate.Year = Date1.Year;
    TempDate.Month = Date1.Month;
    TempDate.Day = Date1.Day;

    Date1.Year = Date2.Year;
    Date1.Month = Date2.Month;
    Date1.Day = Date2.Day;

    Date2.Year = TempDate.Year;
    Date2.Month = TempDate.Month;
    Date2.Day = TempDate.Day;


}
```

```
int GetDifferenceInDays(stDate Date1, stDate Date2, bool
IncludeEndDay = false)
{
    int Days = 0;
    short SawpFlagValue = 1;

    if (!IsDate1BeforeDate2(Date1, Date2))
    {
        //Swap Dates
        SwapDates(Date1, Date2);
        SawpFlagValue = -1;

    }

    while (IsDate1BeforeDate2(Date1, Date2))
    {
        Days++;
        Date1 = IncreaseDateByOneDay(Date1);
    }

    return IncludeEndDay ? ++Days * SawpFlagValue : Days *
SawpFlagValue;
}

short ReadDay()
{
    short Day;
    cout << "\nPlease enter a Day? ";
    cin >> Day;
    return Day;
}

short ReadMonth()
{
    short Month;
    cout << "Please enter a Month? ";
    cin >> Month;
    return Month;
}

short ReadYear()
{
    short Year;
    cout << "Please enter a Year? ";
    cin >> Year;
    return Year;
}
```

```
stDate ReadFullDate()
{
    stDate Date;

    Date.Day = ReadDay();
    Date.Month = ReadMonth();
    Date.Year = ReadYear();

    return Date;
}

int main()
{
    stDate Date1 = ReadFullDate();
    stDate Date2 = ReadFullDate();

    cout << "\nDiffrence is: "
        << GetDifferenceInDays(Date1, Date2) << " Day(s).";

    cout << "\nDiffrence (Including End Day) is: "
        << GetDifferenceInDays(Date1, Date2, true) << " Day(s).";

    system("pause>0");
    return 0;
}
```