a 🡺 If the timer value is stored in a signed **long** 32-bit integer,   
how many **days**, to two decimals, will it take until that integer overflows? **(5 points)**

24hours= 1440mins= 864,00

2,147,483,647/86400/100 = 249 so 249 days

b 🡺 What are the maximum and minimum values that can be stored in a **short** 16-bit signed integer? (2.5 points)

16-bit signed integer maximum = +32,767 … minimum = -32,768

c 🡺 Give examples of values that would cause overflow in positive and negative directions when two **short** 16-bit signed integers are added together. **(5 points)**

32767 + 1 are two positive **short** values causing overflow when added together.

-32,768 + -1 are two negative **short** values causing overflow when added together.

d 🡺 What is potentially wrong with the **(low + high) / 2** calculation to find the middle point? Under what conditions would the calculation go wrong? **(7.5 points)**For a demonstration of the problem, run **MidBugTest.exe** found in this week's zip file.

If I use “(low + high)/2” , I will see errors. The sum of low and high leads to overflow, and then if I devided it by 2, the resulf would be wrong.

e 🡺 REWRITE the code to prevent overflow **(10 points)***from* **mid = (low + high) / 2;***to* mid = low + ( high – low)/2;

f 🡺 Describe the steps you used to develop and test your solution to the binary search bug. **For the full 20 points**, what were the details of your process from problem analysis to solution implementation? (This is like the reflection component in your C course workshop.)

A C program to test your new formula, **MidBugTest.c** is found in this week's zip file. The code demonstrates the bug. Change the **mid =** line of code to your new formula, compile, and run.

The main problem is that the sum of low and high could exceed the maximum value, leading to overflow. I need to avoid overflow. Thus, and integer has an upper bound of 65,535 for an unsigned integer. Consider the low value as 1000 and the high value as 65,000. If I calculate mid as (low + high)/2, it will give me a wrong result due to overflow of integer data type. However, if I put “low + (high – low)/2” condition, it will show a corret result.

**🡺 For any given date, what is the Boolean logic to decide if you have to attend school during the current term?** (a semester is called a “term” at Seneca)

long today, startClasses, endClasses, startStudyWeek, endStudyWeek, startExamWeek, endExamWeek, thanksgivingDay;  
today = 20201007;

startClasses = 20200914;

endClasses = 20201211;

startStudyWeek = 20201026;

endStduyWeek = 20201030;

startExamWeek = 20201207

endExamWeek = 20201211

thanksgivingDay = 20201012

if(today == thanksgivingDay || today >= startStudyWeek && today <= endStudyWeek)

{ hasClass = false;}

elseif

{ classesDow[0],hasClass= true;

classesDow[1], hasClass= true;

classesDow[2], hasClass= true;

classesDow[3], hasClass= true;

classesDow[4], hasClass= false;

classesDow[5], hasClass= false;

classesDow[6], hasClass= flase;

}

else

{hasClass = true ;}

**Activity 3a and 3b** of 3 – **Numbering Systems and Conversions (15 + 10 points)**

3a 🡺 What is the hex value for these colours? **(15 points)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Red decimal** | **Green decimal** | **Blue decimal** | **Hex triplet** | **What  Colour?** |
| 192 | 255 | 238 | #C0FFEE |  |
| 126 | 164 | 112 | #7EA470 |  |
| 186 | 187 | 30 | #BABB1E |  |
| 208 | 13 | 30 | #D00D1E |  |
| 15 | 245 | 231 | #0FF5E7 |  |

3b 🡺 Fill in this chart as per the column headings **(10 points)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **6 digit Hex code** | **Red decimal value (0-255)** | **Green decimal value (0-255)** | **Blue decimal value (0-255)** | **Describe the Final Colour *and* change the cell's background colour, i.e. R-click and see MS Word 'Shading', to match the values for RGB** |
| #D64A53 | 214 | 74 | 83 |  |
| #404893 | 64 | 72 | 147 |  |
| #302430 | 48 | 36 | 48 |  |
| #204C07 | 32 | 76 | 7 |  |