Harshil Prajapati175255215 hprajapati8@mySeneca.ca  
lectures are very intresting

*There are many notes in the instructions to help you earn marks for the questions below.*

Exercise One of Two – **integer overflow** (80 points)

1) 🡺 (7.5 points) If a variable counting hundredths of a second is stored in a signed **long** 32-bit integer,   
how many **days**, to two decimals, will it take until that integer overflows?

It will take 248.55 days to overflow for long 32-bit integer

2) 🡺 (15 points) Convert the maximum value of a signed **long** 32-bit integer, representing hundredths of a second, into whole numbers of  
 days : hours : minutes : seconds . hundredths of a second.   
After *n* days, how many hours remain? After *n* hours, how many minutes remain? etc.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **248.5** | **5,965.2** | **357,913.9** | **21,474,836.4** | **2,147,483,647** |
| **DAYS** | **HOURS :** | **MINUTES :** | **SECONDS .** | **HUNDREDTHS** |

3) 🡺 (2.5 points) What are the maximum and minimum values that can be stored in a **short** 16-bit signed integer?

16-bit signed integer maximum = 32767 … minimum = -32768

4) 🡺 (5+5 points) Give examples of two **short** 16-bit signed integers that when added together would cause overflow.

 15780 + 16989 are two positive values causing overflow when added together.

-17869 + -14989 are two negative values causing overflow when added together.

Binary Search Bug

5) 🡺 (10 points) What is potentially wrong with the **(low + high) / 2** calculation to find the middle point? Under what conditions would the calculation go wrong?

When we add two numbers and output is greater than the capacity of the memory than the airthmatic overflow happens and eventually the calculation goes wrong.

6) 🡺 (10 points) REWRITE themidcalculation to prevent overflow*from*mid = (low + high) / 2;*to*  **mid = (**low + (high-low) ) / 2; **;**

7) 🡺 (25 points)Write a 250+ word “reflection”(similar to a workshop in your programming class) describing the steps you used to develop and test your solution to the calculation bug.

When I open the folder I saw many files in it I read all of that and start understanding the content, where I get to know about the high and low values and also how integer overflows. Then I firstly open the MidBug Test.c in my visual studio 2020 and analys the code and started debugging. step by step I follow all the instruction that are mentioned in the instruction pdf and then I got to know that the sum of low and high integer is greater than the highest maximum integer value than it overflows. At first I don’t get how to solve and debugg properly but after so many attempts I finally get it and I change the formula of mid calculation to mid = (low +(high + low) ) / 2; and then when I run the program I saw no errors and the bug is fixed.

Exercise Two of Two – **Numbering Systems and Conversions (20 points)**

8) 🡺 (10 points ) What is the hex value for these colours?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Red decimal | Green decimal | Blue decimal | Hex triplet | Colour Description |
| 186 | 187 | 30 | BABB1E | Light Yellow |
| 192 | 255 | 238 | C0FFEE | Light Cyan |
| 126 | 164 | 112 | 7EA470 | Pale light Green |
| 15 | 245 | 231 | 0FF5E7 | Sky Blue |
| 208 | 13 | 30 | D00D1E | Red |

9) 🡺 (10 points)Fill in this chart as per the column headings

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Hex triplet | Red decimal | Green decimal | Blue decimal | Describe the Final Colour *and* change the cell's background colour, i.e. R-click and see MS Word 'Shading' |
| #404892 | 64 | 72 | 146 | Dark Blue |
| #D64A52 | 214 | 74 | 82 | Pale Red |
| #204C02 | 36 | 76 | 2 | Dark Green |
| #302437 | 48 | 36 | 65 | Dark Purple |