Answer to Quiz I – Data Structures and Algorithm Analysis [CSC2104]

Sem 3 2017/2018

Question 1 : Given the following declaration:

[6 marks]

int yellow;

unsigned int red;

double green;

* 1. Determine the number of bits allocated for each variable:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| green | **64** | red | **32** | yellow | **32** |

* 1. Give the smallest and largest value can be stored for the following variables:

|  |  |  |
| --- | --- | --- |
|  | yellow | red |
| Highest value | **231-1** | **232-1** |
| Lowest value | **-231** | **0** |

Question 2 - Suppose the following table shows the complexity of 4 different algorithms on solving the same task. Compare and discuss between the algorithm you would most probably choose and another that you would least likely choose.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Algorithm | Complexity | | | |
| Time | | | Space |
| Best | Average | Worst | Worst |
| Algorithm 1 | N | N | N2 | N2 |
| Algorithm 2 | N2 | N2 | N2 | N |
| Algorithm 3 | C | N2 | N3 | N2 |
| Algorithm 4 | N3 | N3 | 2N | N |

[ 4 points ]

**The most likely to be chosen would be Algorithm 2 since compared to others, Algorithm 1 and 2 has the same and most tolerable Big O, but Algorithm 2 has better Space complexity**

**The least desirable should be Algorithm 4 since it has the worst Big O of all the algorithms**

Question 3 - Given the following code:

(4 marks)

void split(int what) {

cout << what <<”split”;

if (what > 1) {

split (what/2);

split (what/2);

}

}

What is the output of the call split (4) from the main module?

**4split2split1split1split2split1split1split**

Question 4 - Given a three-dimensional array called arraydim3 with the dimensions of **2x2x3** is stored based on the *row-wise* manner declared as follows:

short int arraydim3 [2][2][3] = {**7, 2, 16, 28, 110, 52, 81, 13, 55, 72, 19, 11};**

.

* 1. Calculate the address of array element **arraydim3[2, 2, 2]** if it is arranged in column-wise manner when the base address (B) is **123**, size of each element (W) is **2**, and the lowest index of each dimension is **0**.

[ 4 points ]

**123 + 2 [ ((2-1) \*2)/2 + ((2-1)\*2\*2)/2 + ((2-1)\*2\*3)/3]**

**123 + 2 [1 + 2 + 4]**

**123 + 14**

**= 137**

* 1. Rewrite the sequence of the array if it is accessed and displayed in the *column-wise* manner instead.

[ 2 points ]

**7 81 28 72 2 13 110 19 16 55 52 11**