# Softwarica College of IT & E-Commerce

**STW210CT: Programming, Algorithms and Data Structures**Assignment Brief 2022

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Module Name  ST5008CEM: Programming for Developers | Ind/Group  **Individual** | | Cohort  **Nov 2021** | Module Code:  **ST5008CEM** |
| Coursework Title: Coursework | | | | Hand out date: TBD |
| Lecturer: Hikmat Saud | | | | Due date: TBD |
| Estimated Time (hrs):  Word Limit\*: n/a | | Coursework type:  Individual / Practical | | % of Module Mark  33% |
| Submission arrangement online via Softwarica Moodle: Upload through Assignment links  File types and method of recording: URLs (source code repositories) Mark and Feedback date: Within 3 weeks of assignment submission  Mark and Feedback method: Rubric marks and comments via Softwarica LMS | | | | |

|  |
| --- |
| Module Learning Outcomes Assessed:   1. Write software to solve a range of problems. 2. Implement and use simple searching and sorting algorithms. 3. Use libraries to extend the functionality of the base language. 4. Use basic design and testing strategies |
| Notes:   1. You are expected to use the [CUHarvard](https://curve.coventry.ac.uk/open/file/bdfb947c-9d43-48d3-8ec8-f511682e1dd1/1/The%20CU%20Guide%20to%20Referencing%20in%20Harvard%20Style.pdf) referencing format. For support and advice on how this students can contact [Centre for Academic Writing (CAW)](http://www.coventry.ac.uk/study-at-coventry/student-support/academic-support/centre-for-academic-writing/?theme=main). 2. Please notify your registry course support team and module leader for disability support. 3. The University cannot take responsibility for any coursework lost or corrupted on disks, laptops or personal computer. Students should therefore regularly back-up any work and are advised to save it on the University system. 4. If there are technical or performance issues that prevent students submitting coursework through the online coursework submission system on the day of a coursework deadline, an appropriate extension to the coursework submission deadline will be agreed. This extension will normally be 24 hours or the next working day if the deadline falls on a Friday or over the weekend period. This will   be communicated via email and as a Softwarica Moodle announcement. |

# Week 1

# You are provided an array of number and the target value k. explore all possible ways to form an expression by inserting operator such as +, - , \* in a such a way that expression evaluates to the number k.

# Return all expression in the form of string leading to the value k.

# Input: array= {3,4,3}, k=15

# Output: 3\*4+3 and 3+4\*3

# Explanation: 3\*4+3=15 or 3+4\*3=15

**[5 Marks]**

# Week2

You are provided with kth number of linked list, merge all linked list to a single linked list in such a way that at any certain point sum of all values to that point must be positive.

**Input:** list1= {5,7,8,9} list2= {1,2,3,6} list3= {-5, -10,10,11}

**Output:** {1,2,3,5,6,8,9,10,11, -5, -10}

**[5 Marks]**

# Week3

# Given a string of number, return missing smallest positive integer. the time complexity for given algorithm must be 0(n)

# Input: str num=1689

# Output:2

# input: str num=689345

# Output: 1

**[5 Marks]**

# Week4

# your given a coordinate of graph, with x y plane, return length of straight line with maximum number of points on that line.

# Input:{{1,2}, {5,5} {1,4}, {2,3}, {3,2}, {4,1}, {3,5}}

# Output: squrt(18)

**[5 Marks]**

# Week 5

You are given n dimensional array, return matrix with assigned rank to each element of provided array based on below rules.

if x and y exist in same row and column then,

if x>y then rank(x)>rank(y)

if x<y then rank(x)<ran(y)

if x=y then rank(x)=rank(y)

Note that: rank is an integer value greater than its neighboring element and it starts from 1

**Input:** {[2,3], [4,5]}

**Output:** {[1,2], [2,3]}

**[5 Marks]**

# Week6

There are number of application services communicating with each other directly or indirectly with the help of other services.

Return critical path between services that if it crashes then some of the services will not able to communicate with other services making application unable to execute certain requests.

# Input: {{0,1}, {1,6}, {0,3}, {0,2}, {2,3}, {2,5}}

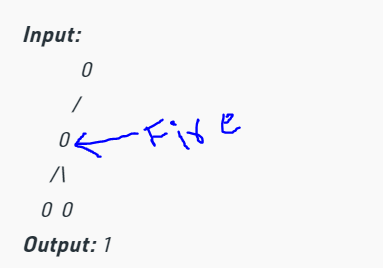
# Output: {2,5} and {1,6}

**[5 Marks]**

**Week7**

# There are N nodes representing the number of houses in the form of tree, return minimum number of nodes in which you can start fire in order to burn all house.

Note that fire starts on any node can burn its children, parents and itself.



**Explanation:** starting fire at level one can burn all other nodes.

**[5 Marks]**

# Week 8

# You are provided with the chemical formula, return maximum number of atoms represented in chemical formula.

# Input: Mg (OH)2

# Output: H2MgO2

# The count of the atom is H:2 0:2 and Mg:1

# [5 Marks]

# Week 8 to 11

# Design and develop GUI application to organize routes information for a transportation company. This application must fulfill below requirements.

# 1) The application should allow to add different destination and their information’s

# 2) Application should store information about connected road link between different destinations with associated cost.

# 3) Application should allow to view different routes between point A to point B

# 4) Application should also display shortest distance between any two points

# Total [60 marks]

# 

# Marking Notes

1. All submitted coursework will be assessed via VIVA conducted at the end of this semester.
2. Each VIVA will last 20 minutes.
3. You will submit on the deadline a document (PDF or Word) on Moodle containing all the coursework tasks solved and including a link to your GitHub Classroom repository shared via Softwarica LMS.
4. During the VIVA you will be assessed with few relevant random questions.
5. If you submit only some of the tasks, your mark will be proportional to that.
6. The marking criteria valid for week 8-11 is presented below.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Criteria** | 0 | 1 | 2 | 3 | 4 | 5 |
| **Feature complete (10)** | Not submitted | Only few features implemented and are not executing | Many of the features are implemented but are not executing correctly | Many of the features are implemented and are executing correctly | Most of the features are implemented and are executing correctly | All features implemented and are executing correctly |
| **Code aesthetic (10)** | Not submitted | Assignment submitted but not commented and formatted. variable’s/classes/function are defined but meaningless | Lack of comments, formatted in Source code. Only few classes and functions are defined but hard to read | Lack of comments, formatted in Source code, but meaningful variable/class/ function names are used few functions are defined. | Lack of comments, formatted in Source code, but meaningful variable/class/ function names are used. Code is easy to read | Source code is well commented, properly formatted, meaningful variable/function/class names are used. Code is easy read and understand, having many pure functions. |
| **GUI (10)** | Not submitted | Hard to use. Only some components are used and unmanaged | Few framed are difficult to use. UI component are used but unmanaged. | Some frames are difficult to use. UI component are used but unmanaged. | Easy to use, Proper use of various UI components. User Interaction is low | Easy to use, Proper use of various UI components, Clean and interactive UI |
| **I/P Validation (10)** | Not submitted | Only few input fields are validated. Error message are not shown | Only few inputs field are validated. Error messages are shown in code format | Most input fields are properly validated. Error messages are shown in code format | Most input fields are properly validated. Error messages are properly shown in natural language | All input fields are properly validated. Error messages are properly shown in natural language. |
| **Unit Testing (10)** | Not submitted | Only few features are tested without using framework and many of them are fail | Many of the modules are tested and many of them are fail | Many of the modules are tested using suitable unit testing framework. | Most of the modules are tested using suitable unit testing framework. Should have partial coverage. | All modules are unit tested using suitable unit testing framework. Should have full testing coverage. |
| **Viva (10)** | Not present (Assignment submitted but absent in viva) | Could not explain reasoning behind the code. But answered only one viva question | Could explain basic terms but not about algorithm. But answered only two viva question | Could explain reasoning behind the code, including use of loops, conditions, algorithms. answered only three viva question | Could explain reasoning behind the code, including use of loops, conditions, algorithms. answered only four viva question | Could explain reasoning behind the code, including use of loops, conditions, algorithms. Answered all five question |