# Comp1 Exam Preparation PackSecret Messages Program

### Aim:

The aim of the document is to help YOU understand the preliminary code and prepare for the COMP1 Exam on 3rd June 2013. This pack specifically prepares you for sections C and D of the Exam

**Section C**

You are advised to spend no more than **20 minutes** on this section. Questions will refer to the **Preliminary Material** and the **Skeleton Program**, but will not require programming.

**Section D**

You are advised to spend no more than **50 minutes** on this section. Questions will use the **Skeleton Program** and the **Preliminary Material**

### Instructions

You need to complete this document on the computer and add rows to the tables below. Feel free to make the document landscape if it helps. Make sure you have saved a copy into your user area BUT beware… I may add things to the tasks below as stuff gets spotted with the code.

1. Complete the table listing all Global variables

|  |  |  |  |
| --- | --- | --- | --- |
| **Variable identifier** | **Data type** | **Line numbers where Used** | **Sample data** |
|  |  |  |  |

1. Complete the table listing all Local variables in the main program,

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Variable identifier** | **Data type** | **Sub routine Identifier** | **Line numbers where used** | **Sample data** |
|  |  |  |  |  |

1. Complete the table listing all Procedures and functions

|  |  |  |  |
| --- | --- | --- | --- |
| **Procedure / function Name and line number** | **Brief description** | **Value parameters** | **Reference parameters** |
|  |  |  |  |

1. Give examples of a variable identifier for each of the following “Variable Roles” .. The \* shows that the role is given as an example in the textbook. This website also has good examples: <http://www.cs.joensuu.fi/~saja/var_roles/>
(whenever I have had rolls from the canteen they have been pretty variable too!)

if that link doesn’t work, try this:
[https://en.wikibooks.org/wiki/A-level\_Computing/AQA/Problem\_Solving,\_Programming,\_Data\_Representation\_and\_Practical\_Exercise/Fundamentals\_of\_Programming/The\_Role\_of\_Variables](https://en.wikibooks.org/wiki/A-level_Computing/AQA/Problem_Solving%2C_Programming%2C_Data_Representation_and_Practical_Exercise/Fundamentals_of_Programming/The_Role_of_Variables)

|  |  |
| --- | --- |
| Role (see page 66 of book or follow links) | Line numbers of examples… Copy and paste code Highlighting the variable being used in that role. |
| [fixed value](http://www.cs.joensuu.fi/~saja/var_roles/role_list.html%22%20%5Cl%20%22fixed) \* |  |
| [stepper](http://www.cs.joensuu.fi/~saja/var_roles/role_list.html#stepper) \* |  |
| [follower](http://www.cs.joensuu.fi/~saja/var_roles/role_list.html#follower) \* |  |
| [most-recent holder](http://www.cs.joensuu.fi/~saja/var_roles/role_list.html#mrh) \* |  |
| [most-wanted holder](http://www.cs.joensuu.fi/~saja/var_roles/role_list.html#mwh) \* |  |
| [gatherer](http://www.cs.joensuu.fi/~saja/var_roles/role_list.html#gatherer) \* |  |
| [one-way flag](http://www.cs.joensuu.fi/~saja/var_roles/role_list.html#owf)  |  |
| [temporary](http://www.cs.joensuu.fi/~saja/var_roles/role_list.html#temporary) \* |  |
| [organizer](http://www.cs.joensuu.fi/~saja/var_roles/role_list.html#organizer)  |  |
| [container](http://www.cs.joensuu.fi/~saja/var_roles/role_list.html#container)  |  |
| [walker](http://www.cs.joensuu.fi/~saja/var_roles/role_list.html#walker)  |  |

1. Complete the table to list all of the selection statements in the code

|  |  |  |  |
| --- | --- | --- | --- |
| **Line Number start** | **Line Number End** | **Type (IF/ ElseIF/Case)** | **Criteria****(treat elseIF as a separate line in this table)** |
|  |  |  |  |
|  |  |  |  |

1. Complete the table to list all of the iteration (looping) statements in the code

|  |  |  |  |
| --- | --- | --- | --- |
| **Line Number start** | **Line Number End** | **Type (For, Do while, Do until)** | **Criteria for exit of loop.** **Fixed🡪 state the number of loops or the stepper variable name****Logic 🡪 state the Boolean expression to exit** |
|  |  |  |  |
|  |  |  |  |

1. List all of the BUILT-IN functions used in the code (not that many!!!)

|  |  |  |
| --- | --- | --- |
| **Line Number**  | **Function identifier** | **Description of usage (include any parameter input and the expected output** |
|  |  |  |
|  |  |  |

1. Complete the table to list all of the places a user has to INPUT a value: Include a description of valid data with examples and invalid data with the name of the error caused.

|  |  |  |  |
| --- | --- | --- | --- |
| **Line Number**  | **User Input Description** | **Vaild data description with examples** | **Invalid data description with the name of the error that will be caused** |
|  |  |  |  |
|  |  |  |  |

1. Complete the table to list all of the places that the code is performing a validation test. )You will have already listed these lines in either the selection or loop statements)

|  |  |  |
| --- | --- | --- |
| **Line Number**  | **Validation Type (see pages 96🡪98 in AS book)**  | **Description of Validation inc. criteria (paste code)** |
|  |  |  |
|  |  |  |

1. Complete the table to produce a simple input test plan

***Erroneous*** – is correct type, but out of range, not on the list etc..
**Invalid** – wrong type of data being entered.
**Normal** – Data the program would expect.
**Boundary – Test the system around the start and end of the data range.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **INPUT Line Number**  | **Description of test** | **Normal Data** | **Erroneous Data** | **Boundary Data** | **Invalid Data** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

1. Complete the table to list all of the places that the code where a validation test is either missing OR incomplete (i.e. the user could enter invalid data causing an error)

|  |  |  |
| --- | --- | --- |
| **Line number of input** | **Validation Type missing** | **Description of Validation inc. criteria NEEDED to be added (include code if you can)** |
|  |  |  |
|  |  |  |

1. Complete the table to list all of the assignment statements in the code where processing occurs, try to explain/describe what is happening to the data during the execution of that line(s) (feel free to put a few lines together.

|  |  |  |
| --- | --- | --- |
| **Line Number**  | **Assignment Statement** | **Explanation** |
|  |  |  |

# Programing Exam Extensions

There are a number of ways in which this code could be extended / improved. Go to the PESTs forum and add any ideas you have. Feel free to add comments to my YouTube Channel for suggestions of videos I could make to help you be successful in this exam.