**Year 12 ATAR Computer Science**

**Task 7 – Programming Production**

**(58 Marks, 5%)**

You have been contracted to create a program for a local patisserie – “For Goodness Cakes”. The program will need to calculate the amount due for individual cake orders. The types of cakes sold include:

Macaroons – individual $ 1.50

Macaroons – 8 pack $ 8.00

Cup cake $ 3.50

Slice $ 3.25

Assorted Box $12.50

* If cake is consumed in the shop, a surcharge of 10% is added to the total amount due.

A function should be used to calculate this amount.

* Take-away cake is charged at the above listed prices.

**Time Allocation**

Two (2) weeks (in and out of class)

**What you need to do**

**Program Development following the SDC**

1. **Requirements analysis (6 marks)**
   1. Make a statement about the requirements of the project as set out by the client. (2 marks)
   2. Outline how you will follow the SDC. (4 marks)
2. **Design data and algorithms (6 marks)**

Develop the algorithms required for the program. These should include the following:

* 1. Pseudocode for program and modules (6 marks)

1. **Code data structures and instructions (8 marks)**
   1. Structure chart for the modules used. (8 marks)
2. **Debug** **syntax and logic errors (7 marks)**
   1. Use trace table to check for errors, using suitable data values (you may use a condensed method) (7 marks)
3. **Test to meet specifications (3 marks)**
   1. Outline how you would test your program prior to implementing it; to ensure it meets specifications (3 marks)
4. **Create code for program (12 marks)**

Using <http://repl.it/>, create:

* 1. Source code for main program, functions and modules (12 marks)

1. **Internal and external documents (5 marks)**
   1. Internal documentation is complete (2 marks)
   2. External document including a user manual and technical manual (3 marks)
2. **Implement/test and evaluate live program (6 marks)**
   1. Explain how you will implement and test your program in the patisserie

(3 marks)

* 1. Explain what you will do to evaluate the program in the patisserie. (3 marks)

**Submission requirements**

|  |  |  |
| --- | --- | --- |
| **What needs to be submitted for assessment** | | **Due date** |
|  | Word processed response to Questions |  |
|  | Algorithm – Pseudocode |  |
|  | Structure Chart |  |
|  | Trace Table |  |
|  | Source code |  |

**Marking Key**

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Description** | **Possible**  **mark** | **Allocated mark** |
| Requirement Analysis  [6] | * Project requirement statement * Outline of SDC | 2  4 |  |
| Algorithms  [10] | * Pseudocode * Use of control structures *(1 mark (each) 1 structure used but not used correct , 2 marks (each) structures documented correctly, 3 marks (each) structures documented and used appropriately)* * Correct use of syntax * Appropriately named variables   *(2 marks each. For each dot point: 2 marks = consistent and correct evidence; 1 mark = inconsistent evidence and 0 marks = no evidence)* | 6  2  2 |  |
| Data structures  [8] | * Structure chart * All modules and functions identified * Correct relationships between modules and functions * All parameters included correctly passed | 3  2  3 |  |
| Debug  [7] | * Trace table * Correct headings * Appropriately selected data values * Tested data values | 2  2  3 |  |
| Test  [3] | * Test   *Outline and justify testing techniques 3 marks*  *Testing techniques with limited justification 2 marks*  *Limited testing techniques mentioned 1 mark* | 3 |  |
| Source code  [12] | **Source code for the application**   * Indenting * Appropriate variable naming  *(2 marks appropriate and consistent, 1 mark for inconsistent)* * Iteration *(1 mark for each instance of using iteration up to a max of 2)* * Selection *(1 mark for each use of selection; up to 2 marks)* * Modules *(1 mark for each module; up to 3 marks correctly used)* * Parameters | 1  2  2  2  3  2 |  |
| Documentation  [5] | **Internal and external documentation**   * Internal   *(consistent commenting 2, inconsistent commenting 1)*   * User documentation | 2  3 |  |
| Implementation/test  [6] | **Implementation**   * Explanation of how you will implement your program * Explanation of how you will evaluate the program | 3  3 |  |
|  |  | **Total 58** |  |