

**Software Design and Development**

**Preliminary Course 2007**

**Programming Assessment Tasks**

**Waverley College**

---

---

## **Introduction**

This document contains all the parts of the preliminary course programming assessment. The task is made up of 10 smaller tasks each of which will take about one week to complete. The tasks are designed to give you a practical introduction to programming for Windows in C# - the best way to learn to program is to do some and that's what this task will do - it also makes up 40% of your preliminary assessment mark.

## **Scheduling**

Each task is designed to take an hour or two to complete and forms part of your normal homework for SDD, each is to be handed in on day H of each of the first 10 cycles (most of terms 1 and 2). This is a little flexible as there are often other activities on around the school, but unless you are specifically told otherwise, every day H there is a task due to be handed in. The first task is due on Friday 9 February.

## **Marking**

Each task is marked out of 5. When all ten tasks have been completed, your total mark will be calculated by taking your best 8 tasks and adding them together to give a mark out of 40. That means that your two worst tasks won't count towards the total, but zero marks given for not handing a task in on time always count. So for example, if you get 5/5 for 4 tasks, 4/5 for 3 tasks, 3/5 for 1 task and don't hand two in within the 3 day limit, your total mark will be  $0 + 0 + 20 + 8 = 28/40$ .

Late hand-ins will result in marks being deducted as follows:

Day A - 1.5 marks deducted

Day B - 3.0 marks deducted

Day C - 4.5 marks deducted

Day D or later - no marks given, the task is still to be handed in.

Normal rules apply for illness or misadventure; you need to fill in the appropriate forms with the required documentation. "My computer is broken" is not an acceptable excuse.

## **Handing in**

Most tasks can be handed in electronically by emailing them to [dreidy@waverley.nsw.edu.au](mailto:dreidy@waverley.nsw.edu.au). In the subject you need to put "SDD Assessment Task" otherwise it may be overlooked and hence not marked. If you want to hand in paper you can either do that in class or at the Science department (not the main staff room). You can't hand in discs, USB memory or other devices; you can hand in iPods with the files, but they won't be returned.

## **Software requirements**

You will require Microsoft Visual Studio 2005 with .Net Framework 2.0 for the tasks. This software is available for free from school, it is your responsibility to make sure you get a copy and get it installed before the first task is due.

## Task 1 – A form

Due \_\_\_\_\_

Requirements:

You are to produce a simple application that displays a single windows form containing:

- A text box
- A label control
- A button

When the button is clicked, the label should display "Hello " and whatever the text in the text box is. So, for example, if "Ruttiger" was typed in the text box and the button clicked, the label would display "Hello Ruttiger".

Marking: (1 mark for each of the following)

- The form
- Having the 3 required elements
- The code works
- Using proper design considerations
- Handling in using the correct procedure

## Task 2 – A simple calculator I

Due \_\_\_\_\_

### Requirements:

You are to produce a simple application that displays a single windows form containing:

- A text box
- A label control
- A button

When the button is clicked, label will show the result of the simple addition you have entered. For example, if you have entered "27 + 14.5" into the text box, the label control will show "41.5". This version of the calculator only needs to do additions with positive numbers.

Marking: (1 mark for each of the following)

- The form with the required elements.
- The code correctly separates the input into numbers and symbols.
- The code works.
- Using proper design considerations.
- Correctly handling integers and real numbers.

### Task 3 – A simple calculator II

Due \_\_\_\_\_

Requirements:

You are to produce a simple application that displays a single window form containing:

- A text box
- A label control
- A button

When the button is clicked, label will show the result of the simple calculation you have entered.

For example, if you have entered "28 \* 3.5" into the text box, the label control will show "98".

This version of the calculator only needs to do calculations with positive and negative numbers and addition, subtraction, multiplication and division. You should modify your program from the last task rather than starting from scratch.

Marking: (1 mark for each of the following)

- The code handles negative numbers correctly.
- The code handles division and multiplication.
- The code handles /0 errors.
- The code works.
- Using proper design considerations.

## Task 4 – List boxes

Due \_\_\_\_\_

### Requirements:

You are to produce a simple application that displays a single windows form containing:

- Two list boxes, one on each side of the form
- Two buttons between the list boxes indicating that one will move items from the left box to the right, and the other from the right box to the left

The program should begin by filling the left hand box with the numbers 1 to 10. Then clicking the buttons should move the selected numbers in one box to the other box. Numbers should be able to be moved both ways with no duplicates ever appearing.

Marking: (1 mark for each of the following)

- The form with the required elements.
- The form initialises the data in the list boxes correctly.
- The buttons move the numbers correctly.
- Using proper design considerations.
- Clicking with nothing selected does not crash the program.

## Task 5 – Editable list boxes

Due \_\_\_\_\_

Requirements:

You are to produce a simple application that displays a single windows form containing:

- Two list boxes, one on each side of the form
- Two buttons between the list boxes indicating that one will move items from the left box to the right, and the other from the right box to the left
- Other controls as needed to achieve the requirements.

This program works in a similar way to the task 4 program, but instead of using numbers, the user should be able to enter items of text into the left list box, once entered, they don't need to be removable. Clicking the buttons should move the selected items in one box to the other box. Items should be able to be moved both ways with no duplicates ever appearing, unless duplicates were entered.

Marking: (1 mark for each of the following)

- The form with the required elements.
- The form allows data entry.
- The buttons move the items correctly.
- Using proper design considerations.
- The program doesn't allow blank data to be added to the list

## Task 6 – File data I

Due \_\_\_\_\_

### Requirements:

You are to produce a simple application that displays a single windows form containing:

- A list view box that displays the data provided in a small .txt file.

The program should fill the list view box (in details mode) with data that is included in a small .txt file. The file contains name, address and telephone data of a few fictitious people. The individual records are separated by newline characters, the individual fields are separated by the | character.

Marking: (1 mark for each of the following)

- The form with the required elements.
- The form opens the data file correctly and retrieves all the data.
- The program separates the lines of data correctly.
- Using proper design considerations.
- The program displays the data correctly

**Task 7 – File data II**

**Due \_\_\_\_\_**

Requirements:

You are to produce a simple application that displays a single windows form containing:

- A list view box that displays the data provided in a small .txt file.
- The required controls for the extra functionality from task 6.

The program should fill the list view box (in details mode) with data that is included in a small .txt file. The file contains name, address and telephone data of a few fictitious people. The individual records are separated by newline characters, the individual fields are separated by the | character. (Use your solution to Task 6 for this.)

In addition to displaying the data it must now be editable. When a user clicks on an item, that item should become editable using other elements on the form. Once edited, the data should display in the list view in place of the original data.

Marking: (1 mark for each of the following)

- The form with the required elements.
- The form fills in initial values when editing is needed.
- The program stores edited information.
- The program correctly displays edited information.
- The program is designed correctly.

## Task 8 – File data III

Due \_\_\_\_\_

### Requirements:

You are to produce a simple application that displays a single windows form containing:

- A list view box that displays the data provided in a small .txt file.
- The required controls for the extra functionality from tasks 6 and 7.

The program should fill the list view box (in details mode) with data that is included in a small .txt file. The file contains name, address and telephone data of a few fictitious people. The individual records are separated by newline characters, the individual fields are separated by the | character.

In addition to displaying the data it must now be editable. When a user clicks on an item, that item should become editable using other elements on the form. Once edited, the data should display in the list view in place of the original data. (Use you solution to task 7 for this.)

There is a “Save” button that allows the user to save their modified data back to disc, overwriting the original .txt file.

Marking: (1 mark for each of the following)

- The program allows the user to choose the storage location.
- The form fills with the new data after it has been saved and the program restarted.
- The program correctly saves the information
- The program handles errors when storing.
- The program is designed correctly.