

# ITECH1000/5000

## Programming 1

School of Science, Information Technology and Engineering

University of Ballarat  
Learn to succeed



Semester2, 2013

### Assignment 1

#### Introduction

A Python program storing and processing simple bank records is in the early stages of development. The records are stored in a text file (**bank.txt**) that contains, for each bank customer, their given name, account number and balance. Each customer record occupies one line in the file and each field is separated by tabs.

Processing of this program is menu driven. Some of the menu items have been coded; others have a stub or dummy code provided. The existing code is provided in a separate Python file (filename: **menu.py**).

#### Assignment conditions

Refer to the course description for the assessment weight of this task, the due time/date and school guidelines in relation to lateness, extensions, special consideration and plagiarism.

#### The report

You will be expected to hand in a report written in Word. Your report should consist of the following items:

1. Section A: answers to the questions.  
Section B: report briefly on each task; for those tasks not attempted report the task letter and the words 'not attempted'.  
Section C: as for section B.  
Section D (Post-graduate students only): as described in the tasks set.
2. Listing of <id>menu.py and <id>.txt where <id> represents your studentID. **Example python file as 30111111menu.py, bank file as 30111111.txt**
3. Acknowledgement: List the name(s) of anyone (other than lecturer or tutor) who assisted you and indicate what assistance they gave. Unless the assistance was substantial it is unlikely to affect the marks awarded, particularly if you were given some explanation and you have been able to write that explanation in your own words. If you did not receive assistance from another person, write 'no assistance received'.

# ITECH1000/5000

## Programming 1

School of Science, Information Technology and Engineering

University of Ballarat  
Learn to succeed



### Assignment submission

Submit your report, modified menu.py and data file. Submit the three files (zipped) to Moodle. The submission to Moodle will be taken as the time of submission.

You are strongly advised to submit your assignment by the due date even if you have not completed every task.

### Tasks

1. Answer the questions in Section A. These are intended to assess understanding of the existing code and the application of general programming principles.
2. Download the text file 'bank.txt'. It should be placed in the same directory as menu.py. Modify the existing code as outlined in Section B.
3. Tackle more demanding problems as outlined in Section C.  
Warning: Section C should only be attempted after all other tasks are completed. For the average student, the time taken for marks gained is much greater in this section. It is intended mainly for the students who are working at an above average level and who enjoy a challenge. However anyone who has made good early progress should benefit.

### Section A - Questions (15 marks)

- a). Run menu.py. Choose option 1 and describe what happens in your report.
- b). Describe briefly the design (structure) of this program.
- c). Line 4 in main() is "opt = menu( )". This is an assignment statement. Explain the various parts of the assignment statement and describe how the right hand side (menu ( )) is evaluated.
- d). In the first option in main( ) there is the line, "displayFile(infile)". Explain the purpose of the variable in the brackets using the terms argument and parameter and referring to the displayFile( ) function as well. You will need to include a description of the relationship between the identifiers "infile" and "inf" in your answer.
- e). Write pseudocode for the task section B, part f) (menu item 8).
- f). What advantages can you identify in using many small functions in this program?
- g). A program takes input, processes and produces output. Assume the user runs 'menu.py' and enters '5'. What are the inputs and expected outputs of this program?

### Section B - Modifications (60 marks)

Modify the program according to the following instructions. (Attempt only those modifications that are specified.) The order given is the order in which the necessary skills are learned. Save your copy of the file using your id in front of menu.py For example, **30089999menu.py**

# ITECH1000/5000

## Programming 1

School of Science, Information Technology and Engineering

University of Ballarat  
Learn to succeed



- a). menu item 3 – the `show_name()` code doesn't work as it does not display the name of the corresponding customer when the correct account number is entered; the `show_mark()` code (item 4) is similar and does work correctly; identify and correct the errors in `show_name()`; report what the errors were and how you fixed them;
- b). menu item 4 – display the result of `show_balance()` in `main()`, not in the function itself; describe the changes you made and explain how they work;
- c). menu item 2 – if an invalid (non-existent) account number is entered, display a suitable message in `main()` (use the comments already written in the function and `main()` as a guide); how useful were the comments?
- d). menu item 7 – code this menu item defining another function using a meaningful name. This function should list all of the balances which are in overdraft (in other words, less than 0). You can use parts of `displayFile()` and `show_balance()` to assist you; what was the most difficult part of this task?
- e). menu item 6 – write function `min_balance()`, based on `max_balance()` to display the lowest balance and also the account number of the corresponding customer(s). There are two customers with the lowest balance. Test the function with both names. Are the results what you expect? Explain.
- f). menu item 8 – write function `find_name()` that displays any record in function that matches the name, and returns the number of customers with that name; display this number in `main()`; what data did you use to test?
- g). menu item 5 – convert the definite loop in `max_balance()` to an indefinite loop with `max` initialised to the balance in the first record; what data did you use to test?

Part marks are available if only some of these tasks are completed successfully, but you should have a working program at all times. Any errors should be confined to individual functions. Using inappropriate names and/or omitting comments may result in marks being deducted (maximum 6 marks in total).

### Section C - Challenge exercises (25 marks)

- a). Use menu item 9 to provide a groupings of customers, displaying only the account of each customer and their group number according to the following rules:

# ITECH1000/5000

## Programming 1

School of Science, Information Technology and Engineering

University of Ballarat  
Learn to succeed



- Group number A. Customers with \$10,000 or more in their accounts
  - Group number B. Customers with \$1000 or more, but less than \$10,000 in their accounts.
  - Group number C. Customers with 0 or more, but less than \$1,000 in their accounts
  - Group number D. Customers in overdraft – less than 0.
- b). Write menu item 10 – allow the user to make a deposit; this will require writing a function that not only reads your text file, identifies the relevant record, and changes the balance, but also writes all records to a temporary file.
- c). An extension of b) - once your text file has been closed in main(), write another function to copy the contents of the temporary file back to the original file. This needs to be conditional on option 9 having been selected.
- d). Write menu item 11 – allow the user to make a withdrawal. This requires basically the same functionality as b) and c) except, of course, money is deducted instead of added. Furthermore, because of a new policy at the bank, customers are no longer allowed to go into overdraft. Thus the program should refuse to allow a withdrawal that is greater than available funds and should print an error message instead.
- e). With the selection of a menu item, only some invalid keystrokes have been dealt with; identify other invalid keystrokes and make corrections to the code so that Python errors are not generated.

# ITECH1000/5000

## Programming 1

School of Science, Information Technology and Engineering

University of Ballarat  
Learn to succeed



### Section D (25 marks)

#### Post-Graduate students (ITECH5000) only – additional task

Write a 500 word article explaining to beginning students how to design programs using top-down and bottom-up approaches. List three advantages of Python language in relation to C++ and JAVA. Include brief descriptions and examples of techniques that contribute to this objective in your article. Consult at least 3 sources. Provide a list of your sources at the end of the article in APA style (refer to your course description for the url of the university's guide). Where you have used ideas either quoting directly or paraphrasing give the in-text reference in APA style.

# ITECH1000/5000

## Programming 1

School of Science, Information Technology and Engineering

University of Ballarat  
Learn to succeed



### Mark Sheet - ITECH1000

Student Name: \_\_\_\_\_ Id: \_\_\_\_\_ Marker: \_\_\_\_\_

Task	Comments	Mark	Max
<b>Section A</b>			<b>15</b>
a)			1
b)			2
c)			2
d)			2
e)			3
f)			2
g)			3
<b>Section B</b>			<b>60</b>
a)			6
b)			6
c)			6
d)			10
e)			8
f)			12
g)			12
<b>Section C</b>			<b>25</b>
a)			5
b)			7
c)			6
d)			3
e)			4
<b>Deductions</b>	comments missing, inappropriate names		max 6
Total(100)			<b>100</b>
<b>Total(20)</b>			<b>20</b>

# ITECH1000/5000

## Programming 1

School of Science, Information Technology and Engineering

University of Ballarat  
Learn to succeed



### Mark Sheet – ITECH5000

Student Name: \_\_\_\_\_ Id: \_\_\_\_\_ Marker: \_\_\_\_\_

Task	Comments	Mark	Max
<b>Section A</b>			<b>15</b>
a)			1
b)			2
c)			2
d)			2
e)			3
f)			2
g)			3
<b>Section B</b>			<b>60</b>
a)			6
b)			6
c)			6
d)			10
e)			8
f)			12
g)			12
<b>Section C</b>			<b>25</b>
a)			5
b)			7
c)			6
d)			3
e)			4
<b>Section D</b>			<b>25</b>
<b>Deductions</b>	comments missing, inappropriate names		max. 6
<b>Total(125)</b>			<b>125</b>
<b>Total(20)</b>			<b>20</b>