

Pearson Edexcel Level 1/Level 2 GCSE

Computer Science

Practical Programming

Controlled Assessment Material

For use from January 2015 to May 2015

Paper Reference

1CP0/02

Data files:

clientIntensity.txt, exerciseActivities.txt, clientRecords.txt,
test_plan.doc

Instructions

- You must complete all tasks.
- You must use a computer to complete these tasks.
- You may **not** have access to the internet.
- You may have access to a printer.
- You may use a calculator.
- You will be given access to relevant files during the controlled assessment.
- All files must be saved and kept secure in the folder specified by your centre between sessions. You will not have access to these files between sessions.
- You must save the final versions of your files in the **CA folder** indicated by your centre.

Information

- The total mark for this controlled assessment is 50.
- The marks for **each** task are shown in brackets.
- The total time for the controlled assessment is 15 hours.
- A guide to how much time you should spend is shown at the end of each task.
- Tasks labelled with an **asterisk** (*) are ones where the quality of your written communication will be assessed
 - *you should take particular care on these tasks with your spelling, punctuation and grammar, as well as the clarity of expression.*

Advice

- Read each task carefully before you start to answer it.
- Attempt every task.

Turn over ►

W47337A

©2015 Pearson Education Ltd.

1/1/1/1/1/1



PEARSON

Controlled assessment tasks

The controlled assessment tasks should be completed using Python, Java or any C-derived language.

The solution must be text-based.

If you use a software development kit such as Xcode or Visual Studio then you must indicate which code you have written.

Context: Exercise Tracker

Daniel Wade is a personal trainer who works with adults aged 19–64. He gives each of his clients an exercise goal to meet each week.

You will complete tasks to produce part of an exercise tracker. The aim of the exercise tracker is to help Daniel record his clients' exercise progress over a period of one week.

Features

- Daniel must enter his clients' details to register them for the exercise tracker.
- A program will generate a client ID, which will be made up of the first 2 letters of the client's first name and the first 3 letters of the client's last name.
- Daniel will record the exercise intensity level of each client.
- Daniel will input the amount of time spent exercising in minutes.
- A feedback program will show the progress of clients.

You have been asked to write programs to:

- register a new client
- allow Daniel to record the activities
- provide feedback on progress.

You must complete the following tasks.

Task	Features	Marks	Suggested Time
1 Registering a new client Program with comments	<ul style="list-style-type: none">• Prompt user to enter registration data	8	1½ hours
2 Creating an activity recorder Design, program, test, evaluate	<ul style="list-style-type: none">• Design a text-based interface• Generate exercise activity• Input activity times	18	5½ hours
3 Creating a feedback program Design, program, test, evaluate	<ul style="list-style-type: none">• Design a text-based interface• Input exercise records from a file• Generate reports	24	8 hours

A summary of the files supplied to you and a checklist of files you should submit is given on page 9.

Task 1: Registering a new client

Daniel will register clients by entering their details. The exercise tracker will generate client IDs for them.

Registration requirements

Here is an outline design for your program.

- Display a welcome message for the exercise tracker.
- Ask the user to enter the client's first name, last name, age, height, weight, and email address.
- Ask the user to choose an exercise intensity level (high or moderate).
- Generate the client ID, which is made up of the first 2 letters of the client's first name and the first 3 letters of their last name.
- Display the client details and client ID and ask the user to confirm they are correct.
 - If any information is incorrect, ask the user to re-enter it then display the client details and client ID again.
- Return to the welcome message.

The client data does not need to be saved. There is no need to validate user input in this task.

Your task

- 1 Write a program to implement this design. It must be easy to read and include comments to explain how it works.

Save the final version of your source code as a file called **registration** in the **CA folder**.

You are advised to spend no more than 1½ hours on Task 1.

(Total for Task 1 = 8 marks)

Task 2: Creating an activity recorder

Daniel must enter the number of minutes each client has spent exercising in one week, for each activity.

clientIntensity.txt contains some client IDs and intensity levels that you may use, or you can create your own. A copy of this is shown in *Appendix A*.

exerciseActivities.txt contains some high and moderate exercise activities that you may use, or you can create your own. A copy of this is shown in *Appendix B*.

Activity recorder requirements

- The user interface for the activity recorder must be text based.
- User instructions must be displayed.
- The user must be asked to enter the client ID of the client they wish to record results for.
- The activity recorder must check the client's intensity level, and then display all relevant activities.
- The user must enter the time spent exercising, between 0 and 120 minutes, for each of the activities. Validation should be used.
- The client ID, along with total time spent exercising in one week, must then be displayed on the screen.
- The user must be asked if they wish to record activities for another client.

The data output from this activity does not need to be saved.

Your task

- 2 (a) (i) Develop a program to meet these requirements. It must be easy to read and include comments to explain how it works. You are expected to decompose your program into subprograms. The quality of your program design will be assessed using the source code you submit.
- (ii) Write and implement a plan to test that the program works correctly and that the requirements have been met. Use the test_plan.doc template provided. A copy of this is shown in *Appendix D*.

Save the final version of your source code as a file called **activity_recorder** in the **CA folder**.

Save your test plan in a file called **activity_test_plan.doc** in the **CA folder**.

*(b) Write an evaluation of your program that:

- assesses how well your program meets the activity recorder requirements
- explains how your program selects which exercise activities to display.

You should use correct technical terms where appropriate.

Save your evaluation in a file called **activity_evaluation.doc** in the **CA folder**.

You are advised to spend no more than 4½ hours on developing and testing the program, and 1 hour on the evaluation.

(Programming and testing = 12 marks)

(Evaluation = 6 marks)

(Total for Task 2 = 18 marks)

Task 3: Creating a feedback program

The exercise tracker must give Daniel feedback on his clients. You will write a program that will enable this to happen.

clientRecords.txt contains the client ID, intensity level, activities, and times for a number of clients. The information is separated by commas and is not ordered. Records are in the format 'client ID, intensity level, Activity1, time, Activity2, time, Activity3, time, Activity4, time, Activity5, time'. A copy of this is shown in *Appendix C*.

Your program must sort and search the file **clientRecords.txt** to select and display information about clients and their exercise progress.

Feedback program requirements

- Four choices must be displayed for the user to select from.

Option A	Activity summary for a client
Option B	Comparison of client progress
Option C	Top performing client
Enter Q to quit	

- The user must be able to select an option or enter 'Q' to quit
- The user interface for the feedback program must be text-based.

Option A must provide a prompt to enter the client ID.

- It must find the exercise times for each activity for the selected client.
- It must calculate the total exercise time for the selected client.
- It must display in this format:

The exercise times for <i>client ID</i> are: Intensity level: <i>intensity level</i>	
Activity	Exercise time (mins)
<i>Activity1</i>	<i>time</i>
<i>Activity2</i>	<i>time</i>
<i>Activity3</i>	<i>time</i>
<i>Activity4</i>	<i>time</i>
<i>Activity5</i>	<i>time</i>
Total time: <i>time</i> mins	

Option B must provide a prompt to enter the intensity level.

- It must find the client ID and exercise times for each activity depending on the selected intensity level.
- It must display the total amount of time spent exercising, for each client.
- It must display in this format. The number of clients may vary.

Comparison of client progress Intensity level: <i>intensity level</i>					
Client ID	<i>Activity1</i>	<i>Activity2</i>	<i>Activity3</i>	<i>Activity4</i>	<i>Activity5</i>
<i>client ID</i>	<i>time</i>	<i>time</i>	<i>time</i>	<i>time</i>	<i>time</i>
<i>client ID</i>	<i>time</i>	<i>time</i>	<i>time</i>	<i>time</i>	<i>time</i>
<i>client ID</i>	<i>time</i>	<i>time</i>	<i>time</i>	<i>time</i>	<i>time</i>
<i>client ID</i>	<i>time</i>	<i>time</i>	<i>time</i>	<i>time</i>	<i>time</i>
<i>client ID</i>	<i>time</i>	<i>time</i>	<i>time</i>	<i>time</i>	<i>time</i>
Total time:	<i>time mins</i>	<i>time mins</i>	<i>time mins</i>	<i>time mins</i>	<i>time mins</i>

Option C must find the top performing client for each intensity level. The top performing client is the client who has exercised for the longest amount of time.

- It must display the clients who have exercised for the longest amount of time at each intensity level. If this is more than one client then each client and their total time should be displayed.
- It must display the amount of time each client has spent exercising.
- It must display in this format:

Top performing clients

Intensity level: High

Client ID has exercised at a high intensity level for *time* minutes this week.

Intensity level: Moderate

Client ID has exercised at a moderate intensity level for *time* minutes this week.

- The feedback program must work for all the records in the **clientRecords.txt** file.

Your task

- 3 (a) (i) Develop a program to meet the above requirements.

It must be easy to read and include comments to explain how it works. The quality of your program design will be assessed using the source code.

- (ii) Write and implement a test plan to ensure that the program works correctly and that the requirements have been met. Use the **test_plan.doc** template provided. A copy of this is shown in *Appendix D*.

Save the final version of your source code as a file called **feedback_program** in the **CA folder**.

Save your test plan as a file called **feedback_program_test_plan.doc** in the **CA folder**.

- *(b) Write an evaluation of your program that:

- assesses how well your program meets the feedback program requirements
- explains what method you used to extract information for each client from the data file.

You should use correct technical terms where appropriate.

Save your evaluation as a file called **feedback_program_evaluation.doc** in the **CA folder**.

You are advised to spend no more than 6½ hours on developing and testing the program, and 1½ hours on the evaluation.

(Programming and testing = 15 marks)

(Evaluation = 9 marks)

(Total for Task 3 = 24 marks)

TOTAL FOR PAPER = 50 MARKS

Files supplied to candidates

Purpose	File name	File type
Client intensity levels	clientIntensity	.txt
Exercise activities	exerciseActivities	.txt
Client records	clientRecords	.txt
Test plan template	test_plan	.doc

Checklist of files for the candidate to submit:

Task	File name	File type
1 program	registration	program source code
2 program	activity_recorder	program source code
2 test plan	activity_test_plan	document
2 evaluation	activity_evaluation	document
3 program	feedback_program	program source code
3 test plan	feedback_program_test_plan	document
3 evaluation	feedback_program_evaluation	document

Appendix A

You have been given the file **clientIntensity.txt**. A copy of the file is shown here.

The data is shown as: 'client ID, intensity level'.

NeQua,High
ImKol,Moderate
YoTri,Moderate
RoDen,High
NaThe,Moderate
ReWes,Moderate
BrFre,High
KaDat,High
ViRil,High
TrGeo,High

Appendix B

You have been given the file **exerciseActivities.txt**. A copy of the file is shown here.

High

Running

Swimming

Aerobics

Football

Tennis

Moderate

Walking

Hiking

Cleaning

Skateboarding

Basketball

Appendix C

You have been given the file **clientRecords.txt**. A copy of the file is shown here.

The data is shown as: 'client ID, intensity level, Activity1, time, Activity2, time, Activity3, time, Activity4, time, Activity5, time.'

NeQua,High,Running,5,Swimming,40,Aerobics,40,Football,20,Tennis,10
ImKol,Moderate,Walking,40,Hiking,0,Cleaning,40,Skateboarding,30,Basketball,20
YoTri,Moderate,Walking,20,Hiking,30,Cleaning,40,Skateboarding,20,Basketball,40
RoDen,High,Running,20,Swimming,20,Aerobics,40,Football,30,Tennis,50
NaThe,Moderate,Walking,30,Hiking,30,Cleaning,20,Skateboarding,10,Basketball,30
ReWes,Moderate,Walking,30,Hiking,20,Cleaning,50,Skateboarding,40,Basketball,20
BrFre,High,Running,20,Swimming,30,Aerobics,30,Football,30,Tennis,20
KaDat,High,Running,30,Swimming,20,Aerobics,10,Football,20,Tennis,30
ViRil,High,Running,50,Swimming,50,Aerobics,60,Football,40,Tennis,50
TrGeo,High,Running,10,Swimming,20,Aerobics,30,Football,30,Tennis,20
DaWay,High,Running,60,Swimming,50,Aerobics,40,Football,50,Tennis,50
CaAma,High,Running,30,Swimming,20,Aerobics,10,Football,20,Tennis,30
ArRes,High,Running,0,Swimming,10,Aerobics,30,Football,20,Tennis,20
BeVic,High,Running,20,Swimming,20,Aerobics,30,Football,30,Tennis,10
MaFre,High,Running,10,Swimming,20,Aerobics,20,Football,40,Tennis,30
AnMer,Moderate,Walking,40,Hiking,30,Cleaning,30,Skateboarding,20,Basketball,30
SaBro,Moderate,Walking,30,Hiking,20,Cleaning,30,Skateboarding,20,Basketball,20
PoLig,Moderate,Walking,20,Hiking,20,Cleaning,20,Skateboarding,30,Basketball,30
HeZbe,Moderate,Walking,30,Hiking,40,Cleaning,20,Skateboarding,40,Basketball,10
GiLop,Moderate,Walking,40,Hiking,0,Cleaning,30,Skateboarding,40,Basketball,20
DeTur,Moderate,Walking,10,Hiking,0,Cleaning,10,Skateboarding,30,Basketball,30
LaKin,Moderate,Walking,20,Hiking,20,Cleaning,30,Skateboarding,30,Basketball,20
AnVen,Moderate,Walking,50,Hiking,0,Cleaning,50,Skateboarding,50,Basketball,20
LoLew,Moderate,Walking,10,Hiking,20,Cleaning,10,Skateboarding,30,Basketball,40
NyRed,Moderate,Walking,0,Hiking,0,Cleaning,0,Skateboarding,0,Basketball,20

Appendix D

You have been given the file **test_plan.doc**. A copy of the file is shown here.

Test number	Description of test	Test data	Expected result	Actual result	Pass/Fail