



**Cambridge Assessment International Education**  
Cambridge Ordinary Level

---

**COMPUTER SCIENCE**

**2210/21**

Paper 2 Problem-solving and Programming

**May/June 2019**

PRE-RELEASE MATERIAL

No Additional Materials are required.

**This material should be given to the relevant teachers and candidates as soon as it has been received at the centre.**

---

**READ THESE INSTRUCTIONS FIRST**

Candidates should use this material in preparation for the examination. Candidates should attempt the practical programming tasks using their chosen high-level, procedural programming language.

Any businesses described in this paper are entirely fictitious.

---

This document consists of **2** printed pages.

In preparation for the examination candidates should attempt the following practical tasks by **writing and testing a program or programs**.

The local high school uses buses to transport students to school. There are six bus routes labelled A to F. You have conducted a survey to analyse the punctuality statistics of these buses over a four-week period. The data from the survey are shown in the table:

	Punctuality table					
Day	Bus A	Bus B	Bus C	Bus D	Bus E	Bus F
Mon1	0	0	2	1	-1	0
Tue1	0	1	0	0	-1	-5
Wed1	0	0	-1	0	-1	-5
Thu1	2	0	-1	0	-2	-5
Fri1	2	1	-2	0	-4	-4
Mon2	4	2	-2	0	-10	-3
Tue2	0	0	-3	0	-2	-5
Wed2	3	0	-1	0	0	0
Thu2	4	0	0	0	0	0
Fri2	-2	0	0	0	0	0
Mon3	-5	1	-2	2	0	0
Tue3	0	0	0	0	1	-2
Wed3	0	0	1	0	2	-3
Thu3	3	0	1	0	-3	1
Fri3	4	2	1	0	1	1
Mon4	-1	0	1	0	1	1
Tue4	8	0	-1	0	3	0
Wed4	1	1	-1	0	-1	0
Thu4	1	0	2	0	0	-2
Fri4	-2	0	-2	0	0	-5

Positive numbers represent minutes early, negative numbers represent minutes late and 0 represents the bus having been on time.

Write and test a program or programs for the local high school.

- Your program or programs must include appropriate prompts for the entry of data; data must be validated on entry.
- Error messages and other output need to be set out clearly and understandably.
- All variables, constants and other identifiers must have meaningful names.

You will need to complete these **three** tasks. Each task must be fully tested.

### Task 1 – Setting up the data storage.

Using arrays set up a system to enable data for each bus route to be entered covering each day of a four-week period. It must be possible to enter the data supplied or your own set of data, using suitable prompts as necessary.

### Task 2 – Working out the statistics.

Extend your program so that the following statistics for the four-week period may be calculated and output:

- the number of late arrivals for each bus route
- the average number of minutes late for each bus route
- the bus route with the highest number of days on which it was late
- the average number of minutes late for each bus route, using only data from days on which it was late

All the results should be displayed with appropriate annotation.

### Task 3 – Checking specific days.

Extend the program as follows:

- Allow the user to input a specific day, for example Fri3, to be used for analysis of data.
- Find and display how many buses were late on this particular day.
- For each late bus, display the route label and how late the bus was on this particular day.