

Coursework Task

Higher Computing Coursework Task 2009–2010

Part 1

Click-Kit, a company which sells photographic equipment, decides to commission a software development company to produce a stock control program that will:

- store the names of items held in stock
- calculate product codes of held items
- produce a list of products stocked, with their initial stock levels
- allow information on products held in stock to be found
- allow products to be purchased
- produce a list of final stock levels, indicating which items need to be re-stocked.

The names of five digital cameras and their stock levels which should be used in testing the program are given below.

Product Name	Stock Level
Simpsun GN120	1
Sonic Lux10	2
Ultimax G42	4
Antalpha A200	3
Nickov N230	3

You may use any suitable method available in your programming environment to enter this data into the system.

How the program should work

Storing the Product Names

The program should take in the product names and initial stock levels. It should store them in a suitable data structure.

Calculating and Storing the Product Codes

The program should calculate the product codes using the first three and last three characters of the product name. It should store them in a suitable data structure.

Displaying the Initial Stock List

The program should display a list of the product names, their product codes and the initial stock levels (see page 8 for an example layout).

Displaying a Menu

The program should then offer the user the following three menu options:

- **Finding a Product**

The program should ask the user for a **product code** then find the product. If the product code is **not** found then an appropriate message should be displayed. Otherwise it should display the product name, product code and stock levels.

- **Purchasing an Item**

The program should ask the user to enter the product code of an item. If the product is **not** in stock a suitable error message should be displayed. Otherwise a suitable message should be displayed confirming it is in stock and confirming the purchase. The stock level for that item should then be decreased by one.

Note: this part of the program should be tested by the user purchasing a **Sonic Lux10** and **Nickov N230**.

- **Quit**

The program should display a suitable message.

Displaying the Final Stock List

After quit is chosen, the program should display a list of the product names, their product codes and the final stock levels. The program should also display a re-order message for each product where the number in stock is less than two.

Displaying the Results

The output from the program should be in columns, similar to the examples shown below.

- The **initial stock list** should look something like this . . .

Product Name	Product Code	Initial Stock Level
Simpsun GN120	Sim120	1
Sonic Lux10	Sonx10	2
Ultimax G42	UltG42	4
Antalpha A200	Ant200	3
Nickov N230	Nic230	3

- The output of **finding a product** should look something like this . . .

Product to be found: UltG42

Product Name	Product Code	Stock Level
Ultimax G42	UltG42	4

- The output of **purchasing a product** should look something like this . . .

Product to be purchased: Sonx10

Product Name	Product Code	Status
Sonic Lux10	Sonx10	In stock
Purchase confirmed		

- The **final stock check** should look something like this . . .

Product Name	Product Code	Stock	Action
Simsun GN120	Sim120	1	Re-order
Sonic Lux10	Sonx10	1	Re-order
Ultimax G42	UltG42	4	
Antalpha A200	Ant200	3	
Nickov N230	Nic230	1	Re-order

Algorithm

1. Enter and store product names and initial stock levels
2. Calculate and store product codes
3. Display product names and codes
4. Start conditional loop
5. Display menu
6. Get option from user
7. Where option is F, perform Find a product
8. Where option is P, perform Purchase a product
9. Where option is Q, perform Quit
10. End conditional loop when Q is chosen
11. Display final stock check

What you have to do:

Tasks		Evidence required	Marks
1	Indicate data flow on the algorithm.	Algorithm with data flow.	3
2	Refine steps 2, 7 and 8 of the algorithm.	Pseudocode for steps 2, 7 and 8.	7
3	Using a software development environment of your choice, implement the algorithm. Use separate sub-programs where appropriate. Use parameter passing where appropriate.	Listing of implemented program.	16
4	Test the program with the data provided to ensure that it is fit for purpose.	Hard copy of test results.	1
5	Evaluate the test results.	Brief report on test results.	3

Marking Grid

Name _____

Date _____

Topic		Out of	Mark	Comment
Part 1				
Design (10)	Indication of data flow	3, 2, 1, 0		
	Pseudocode for step 2	2, 1, 0		
	Pseudocode for step 5	2, 1, 0		
	Pseudocode for step 6	3, 2, 1, 0		
Implementation (16)	Main program	2, 1, 0		
	Sub-program for step 2	2, 1, 0		
	Sub-program for step 5	2, 1, 0		
	Sub-program for step 6	3, 2, 1, 0		
	Formatted output	2, 1, 0		
	Use of parameters	3, 2, 1, 0		
	Maintainability	2, 1, 0		
Correcting errors (4)	Testing	1, 0		
	Evaluating fitness for purpose	1, 0		
	Evaluating maintainability	2, 1, 0		
Part 2				
Identify and justify mini desktop (8)	Identify two suitable mini desktops	2, 1, 0		
	Compare mini desktops according to processor speed, backing storage capacity, RAM capacity, and power consumption	4, 3, 2, 1, 0		
	Recommend one mini desktop using the above criteria and justify the recommendation in terms of the context	2, 1, 0		
Identify and justify a tape backup drive (7)	Identify two suitable tape drives	2, 1, 0		
	Compare two tape drives in terms of buffer capacity, cartridge capacity and data transfer rate	3, 2, 1, 0		
	Recommend one tape drive using the above criteria and justify the recommendation in terms of the context	2, 1, 0		
Identify and justify a wireless colour laser (7)	Identify two suitable networked colour laser printers	2, 1, 0		
	Compare two networked colour laser printers in terms of RAM capacity, print resolution and print speed	3, 2, 1, 0		
	Recommend one laser printer using the above criteria and justify the recommendation in terms of the context	2, 1, 0		
Identify and justify data recovery software (6)	Identify two suitable data recovery software packages	2, 1, 0		
	Compare the recovery software packages in terms of two suitable features	2, 1, 0		
	Recommend one data recovery package using the above criteria and justify the recommendation in terms of the context	2, 1, 0		
Overall report (2)	The total cost of all recommended hardware is within the £8500 budget	1, 0		
	Completeness and clarity of report	1, 0		
Overall total		60		

Allocation of marks

Award full marks if achieved successfully without assistance.
 Award less than the maximum if task achieved **partially** without assistance.
 Award one mark if completed with **some** assistance or hints.
 Award no marks if item not achieved, or completed only with **significant** assistance.

Further Guidelines for Teachers/Lecturers

(Not to be distributed to candidates)

Part 1

It may be necessary, in some programming environments, to introduce a “step zero” into the algorithm to initialize the variables.

The supplied data can be input into the program in any suitable fashion ie typed in by the user, read from data statements or even a file. Pupils are not required to fully test for robustness.

Purchase an item: this part of the program should be tested by the user purchasing a **Sonic Lux10** and **Nickov N230** in order to produce the final stock check.

When evaluating maintainability candidates should consider the use of:

- white space – indentation and blank lines
- meaningful identifiers eg procedure, variable names
- internal commentary.

Two marks should be awarded if the evaluation refers to all three of the above bullet points. One mark should be awarded if the evaluation refers to two of the above. No marks should be awarded if the evaluation refers to fewer than two of the above. It can be assumed that the item names are a minimum of six characters long.

Part 2

Only the base units for the mini desktops are needed, Click-Kit already has ten adequate TFT monitors, keyboards and mice.

Suggested dimensions of mini desktops are for guideline only.

Data recovery software may be one package, a suite of packages or a combination of several distinct packages. A site licence may be the most economical solution.

Suggested suitable features for comparison of data recovery software are range of file types recoverable and the range of types of media from which data can be recovered.

The tape drive may be external or internal. If the latter, then compatibility can be assumed.