University of Hertfordshire UH
School of Physics, Engineering and Computer Science

Assignment Briefing Sheet (2024/25 Academic Year)

Section A: Assignment title, important dates and weighting

Assignment title:	A2 - Inpatient Checkout Simulator in C++	Group or individual:	Individual
Module title:	Object Oriented Programming	Module code:	5FTC1330 & 5FTC1332
Module leader:	Ahmed Bin Mazhar	Moderator's initials:	
Submission deadline:	**27/04/2025**	Target date for return of marked assignment:	

You are expected to spend about 70 hours to complete this assignment to a satisfactory standard.

This assignment is worth **70%** of the overall assessment for this module.

Section B: Student(s) to complete

Student ID number	Year Code

Notes for students

- For undergraduate modules, a score above 40% represent a pass performance at honours level.
- For postgraduate modules, a score of 50% or above represents a pass mark.
- Late submission of any item of coursework for each day or part thereof (or for hard copy submission only, working day or part thereof) for up to five days after the published deadline, coursework relating to modules at Levels 0, 4, 5, 6 submitted late (including deferred coursework, but with the exception of referred coursework), will have the numeric grade reduced by 10 grade points until or unless the numeric grade reaches or is 40. Where the numeric grade awarded for the assessment is less than 40, no lateness penalty will be applied.
- Late submission of referred coursework will automatically be awarded a grade of zero (0).
- Coursework (including deferred coursework) submitted later than five days (five working days in the case of hard copy submission) after the published deadline will be awarded a grade of zero (0).
- Regulations governing assessment offences including Plagiarism and Collusion are available from https://www.herts.ac.uk/about-us/governance/university-policies-and-regulations-uprs/uprs (please refer to UPR AS14)
- Guidance on avoiding plagiarism can be found here: https://herts.instructure.com/courses/61421/pages/referencing-avoiding-plagiarism?module_item_id=779436
- Modules may have several components of assessment and may require a pass in all elements. For further details, please consult the relevant Module Handbook (available on Studynet/Canvas, under Module Information) or ask the Module Leader.

Assignment Briefing Sheet (2024/25 Academic Year)

This Assignment assesses the following module Learning Outcomes (from Definitive Module Document):

- LO1 – Distinguish basic object-oriented programming concepts and methodologies
- LO2 – Examine how these concepts and methodologies are used to create programs
- LO3 – Design, implement, test and debug simple programs in an object-oriented language
- LO4 – Apply basic object-oriented design methodologies to develop object-oriented programs from a program specification

Assignment Brief:

Inpatient Checkout Simulator

NO AI: AI tools must not be used at any point during the assessment. Learner must complete the assessment entirely without AI assistance and rely solely on their knowledge understanding and skills.

Requirements:

Develop a C++ project that computes a patient's bill for a hospital stay. The project should include a *PatientAccount* class, a *Surgery* class, and a *Pharmacy* class.

- The ***PatientAccount*** class should keep track of the total number of days the patient has spent in the hospital, and computes the total charge for the entire stay (given that the hospital's daily room rate is £200). This class will also add any additional charges to the patient's bill if fees have been incurred for any surgery or medication.

- The ***Surgery*** class contains a record of charges for different types of surgeries, as shown below. The class should also print a user menu to allow the user to query a particular charge.

Cataract surgery	£1500
Debridement	£180
Low back pain surgery	£900
Tonsillectomy	£2100

- The ***Pharmacy*** class contains a record of prices for different types of medication, as shown below. It also prints a user menu to allow the user to query prices.

Antibiotic	£20
Anti-inflammatory	£17
Anti-nausea	£38
Inhalant	£66
Pain Relief	£35

The program should prompt the user to input the number of days that a patient has stayed in the hospital, the type of surgery that has been carried out (if any), the medication (if any), and print out the total bill for that patient.

Sample Output:

When the program starts, the system should print a simple greeting and prompt the user to enter the number of days that the patent has stayed in the hospital. A sample run is:

```
**********************
Inpatient Checkout Menu
**********************

How many days was the patient in the hospital? 4
```

Next, the user can enter any surgery that has occurred during the hospital stay, if any. A sample run is:

```
----------------------------------------
Which type of surgery has been performed?
0. None
1. Cataract surgery
2. Debridement
3. Low back pain surgery
4. Tonsillectomy
----------------------------------------
Choose a type of Surgery (0-4):  2
```

And a similar process should take place for medication. A sample run is:

```
----------------------------------------
Which follow-up medication has been given?
0. None
1. Antibiotic
2. Anti-inflammatory
3. Anti-nausea
4. Inhalant
5. Pain Relief
----------------------------------------
Choose a type of medication (0-5):  4
```

Finally, the program prints the total bill. A sample run can be:

```
Total bill for the patient:  £1046.00
```

The program should repeat the above process for any subsequent patients, until the user types ‘n’ or ‘N’ to exit the program.

```
Next patient? (Y/N) y
```

There should basic error checking for surgery and medication entries, for example:

```
Choose a type of Surgery (0-4):  55

Please only enter a number between 0 - 4:
```

Demo Video:

A demo video can be found on Canvas, serves as guidance for development.

Design Strategy:

There may be many approaches to implement this program. Here is one possible pathway for the development of core elements:

1. Write a simple `main` file. Define simple classes for all the classes, which only contain member variables. (You can address member functions later.)
   (Refer to Lecture 5 and Lab 5.1 for how to define classes and declare objects.)
   (Refer to Lecture 4 and Lab 4 for using arrays.)

2. Separate classes from the `main` program and save them as independent header files.
   (Refer to Lecture 5, Lab 5.2, and Lab 5.3 for the file protocol.)

3. Define a set of constructors and destructors for all existing classes.
   (Refer to Lecture 8, Lab 8.1, and Lab 8.2 for how to define constructors and destructors.)

4. Define a set of mutators and assessors (if applicable) for all the existing classes.
   (Refer to Lecture 7, Lab 7.1, and 7.2 on how to define mutators and assessors as member functions.)

5. Build other member functions into existing classes where necessary.
   (Watch the demo and read the assignment requirements again to see what is needed in each class.)

6. Add overloading constructors where necessary.
   (Refer to Lecture 8 and Lab 8.3 for overloading constructors.)

7. In the `main` program, implement suitable program control (e.g. apply a loop to control the system operation), as well as the initial menu, greeting, and printing of final bill. (Refer to Lecture 1 – 3 and Lab 1 – 3 for I/O and decision-making in programs.)

8. Improve code efficiency and robustness as needed.
   (Revise the whole program overall. Refer to Lecture 10 for general guidelines of advanced programming paradigms. Make sure appropriate data types are chosen. Double-check access permissions for all class members. Add clear and suitable comments to the code to document key features, processes, and decisions.)

Further Development:

If you wish to extend this project a further to increase depth and functionality (you will be given credit for this, as outlined in the marking scheme below), you can consider including additional features, for example:

- Allow the user to enter the patient's name.
- Allow the user to choose a quantity for each medication, along with multiple medications and surgeries.
- Print a detailed and formatted final bill, detailing a full list of costs incurred.

Note that the design of these additional features should also follow the good OOP practice, for example, it may not be advisable to put added features of code within the `main`, but rather to include in parts of the relevant classes.

Demonstration (Multiplier) Date to be confirmed:

You are also required to demonstrate your program to your tutor. You are required to demonstrate all the functionality of your program and explain your code. You may also be asked questions. The demonstration acts as a multiplier and rewards more of your assignment grade based on how well you do. Unable to explain the OOP principles will affect your overall grade in other tasks as well. Please see the mark breakdown for more information.

A grade 5 will be awarded to a student who can demonstrate their program, can answer all questions correctly and can explain how their program functions in detail.

Submission Requirements:

This is assignment is to be submitted and marked anonymously. Students should ONLY use their student ID number to identify themselves on their work. Work submitted via Canvas for anonymous marking will automatically have an anonymity number allocated to it.

All submission should be made via Canvas, including:

- One zipped file for all program files.

Marks awarded for:

Components	Marks
OOP Programming Technique	60%
Class definition and File protocol — *6%* *Appropriate data types and access permissions* — *6%* *Default constructor and destructor* — *6%* *Overloading constructors* — *6%* *Mutators and accessors* — *6%* *Main program* — *5%* *Program control techniques* — *5%* *Extended features, if any* — *10%* *Appropriate comments on the code* — *10%*	
Program Execution	40%
Initial menu print — *5%* *Correct calculations for the number of days* — *5%* *Correct calculations for surgery type* — *5%* *Correct calculations for medication type* — *5%* *Clear and accurate printing of final bill* — *5%* *Extended features, if any* — *10%* *User-friendly interface and smooth operation* — *5%*	
Total	**100%**
Demonstration multiplier	**/5**

Demonstration Grade	Multiplier
5/5	100%
4/5	70%
3/5	60%
2/5	50%
1/5	40%
0/5	0%
No demonstration	0%

Final Mark	**Total x Multiplier**

Type of Feedback to be given for this assignment:

Each student will receive written feedback on this assignment, as well as marks awarded based on the criteria set out above.

University of Hertfordshire UH

School of Physics, Engineering and Computer Science

Numeric Grade					
		Marks or %	***Marks or %***	***Marks or %***	***Marks or %***
90-100	**Outstanding**	Outstanding breadth and depth demonstrated. Outstanding integration of literature and/or theory into work. Outstanding exploration and demonstration of topic showing in depth knowledge and understanding	Outstanding use of appropriate technologies as applied to the problem domain. Occasionally stepping beyond expectations using sophisticated solutions. Consistently accurate and outstanding application of skills and techniques demonstrated	Solutions are novel within context. Profound depth of engagement and incisive ideas development. Planning is meticulous; decision making is perceptive. Methodologies are used in an outstanding manner.	Outstanding level of analysis, critical evaluation and/or reflection with outstanding application to derived solutions (where required). Highly developed / focused work. Original and well-informed personal response
80-90	**Excellent**	Excellent breadth and depth demonstrated. Excellent integration of literature and/or theory into work. Excellent exploration and demonstration of topic showing in depth knowledge and understanding	Excellent use of appropriate technologies as applied to the problem domain. Occasionally stepping beyond expectations using sophisticated solutions. Consistently accurate and Excellent application of skills and techniques demonstrated	Solutions reframe task within context. Profound depth of engagement and incisive ideas development. Planning is meticulous; decision making is perceptive. Methodologies are used in an Excellent manner	Excellent level of analysis, critical evaluation and/or reflection with Excellent application to derived solutions (where required). Highly developed / focused work. Original and well-informed personal response
70-79	**Very good**	Very good breadth & depth demonstrated. Very good integration of literature and/or theory into work. Very good level of knowledge and understanding demonstrated. Covers all relevant points and issues.	Very good use of appropriate technologies as applied to the problem domain. Very good and highly accurate application of skills and techniques demonstrated. Minor errors in technique and/or application with little or no impact on deliverables.	Solutions are innovative within its context. Considerable depth of engagement and successful ideas development clearly documented. Detailed planning and clear rationale for decisions. Methodologies are used in a very good manner	Very good level of analysis, critical evaluation and/or reflection of issues with Very good application to derived solutions (where required). Well-developed personal response
60-69	**Good**	Good breadth & depth demonstrated appropriate to topic. Literature and/or theory integrated very well. Good level of knowledge and understanding demonstrated.	Good use of appropriate technologies as applied to the problem domain. High level and very accurate application of skills and techniques understanding demonstrated. Small errors in technique and/or application with little impact on deliverables	Solutions relate directly to task and may step beyond conventions Strong engagement with subject material and processes, evaluation of alternatives, solutions come from process. Strong use of methodologies to derive solutions	good level of, analysis, critical evaluation and/or reflection but not consistently taken to full extent with good application to derived solutions (where required). Partial personal response tends towards descriptive
50-59	**Clear Pass**	Good use of legends. Depth appropriate to topic BUT moderate breadth or vice versa. Literature and/or theory integrated into work. Good grasp of the topic and some of its implications. Knowledge and understanding is demonstrated. Minor errors / omissions.	Evidence of use of appropriate technologies as applied to the problem domain. Good and reasonably accurate application of skills and techniques demonstrated. Some errors in technique and/or application with minor impact on deliverables	Solutions are appropriate to task, work well within conventions. Evidence of use of subject material and planning processes, range of alternatives put forward and evaluated Experimentation to support implementation within conventions Good use of methodologies to derive solutions	Evidence of analysis and/or reflection but critical evaluation could be expanded on further. Good application to derived solutions (where required). Primarily descriptive personal response, sometimes restricted to immediate concerns
40-49	**Marginal Pass**	Satisfactory use of legends demonstrated but limited in breadth OR depth. Uncritical and quoted without comment where necessary. Satisfactory content / level of knowledge of the topic. Addresses part of the question. Some errors / omissions.	Satisfactory use of appropriate technologies as applied to the problem domain. Satisfactory application of skills and techniques demonstrated but with minor inaccuracies. Errors in technique and/or application with some impact on deliverables	Solutions limited to task and address conventions. Solutions found or adopted Some planning but completion is rushed. Some experiments but limited alternatives. Methodologies are applied to derive solutions but steps are missed.	Satisfactory level of analysis and/or reflection but limited evidence of critical evaluation. Satisfactory application to derived solutions (where required). Descriptive personal response mainly restricted to immediate concerns

30-39	Marginal Fail	Limited in breadth and depth demonstrated. Some parts of legends used/quoted without comment. Limited content / knowledge. Limited or muddled understanding of the topic/question.	Limited use of appropriate technologies as applied to the problem domain. Limited application of skills and techniques demonstrated. Many errors in technique and/or application with high impact on deliverables.	Solutions reframe task inappropriately and do not address conventions. Basic use of strategies, few alternatives, limited evaluation with limited experimentation. Limited use of methodologies to derive solutions.	Limited evidence of analysis, critical evaluation and/or reflection. Limited application to derived solutions (where required).Too descriptive in parts. Limited personal response.
20-29	Clear Fail	Lacking in breadth and depth. Some parts of legends irrelevant to topic area. Lacking knowledge Content irrelevant / inaccurate. Does not address the question and therefore does not meet the learning outcomes.	Very little use of appropriate technologies as applied to the problem domain. Very little skill and application of techniques demonstrated. High number of errors with very high impact on deliverables.	Lacking in appropriate solutions with very limited use of strategies, no evaluation and little evidence of ideas development. Little use of methodologies	Lacking in its level of analysis / critical evaluation and/or reflection. Minimal application to derived solutions (where required) Mainly descriptive, lacking in personal response.
0-19	Little or Nothing of Merit	No / unsatisfactory evidence of Legends used and irrelevant to topic area No / unsatisfactory level of knowledge demonstrated. Content not appropriate to the topic.	No use of appropriate technologies as applied to the problem domain. No skill and application of technique demonstrated. Very high number of errors in deliverable or no deliverable submitted.	No or completely inappropriate solution. No use of strategies, no planning and no experimentation. No application of methodology	Unsatisfactory level of analysis / critical evaluation and or reflection. No application to derived solutions (where required) Wholly descriptive. No personal response.