

CSYM 021 Java Programming

Assignment:

Development of 2D/3D Graphics Software: (100%).

Due for Issue (week commencing):	Tuesday, 26 February 2009	Date for Submission:	Friday, 30 May 2009
---------------------------------------------	------------------------------	---------------------------------	------------------------

(Friday: Hand into Student Assessment Office (Open 10:00 - 15:00)).

Aim: Produce a technical report and accompanying application using Java 2D/3D.

Brief

Produce a technical report and accompanying application using Java 3D. The application is to be a Java 3D environment in the simplified form of the University of Northampton, Kingsley Building to include the Ground, First and Second floors. The user should be able to move horizontally along the corridor and into the rooms on at least the 1st Floor. The South-East Front Elevation is shown in **Figure 1** and the 1st Floor plan is shown in **Figure 2**:

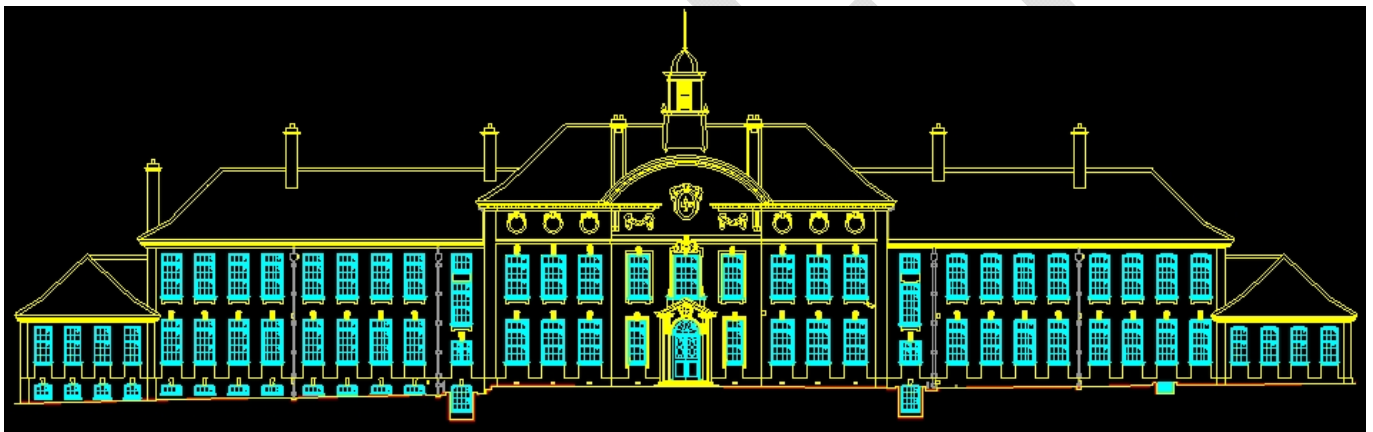


Figure 1: Kingsley Building – South-East Front Elevation

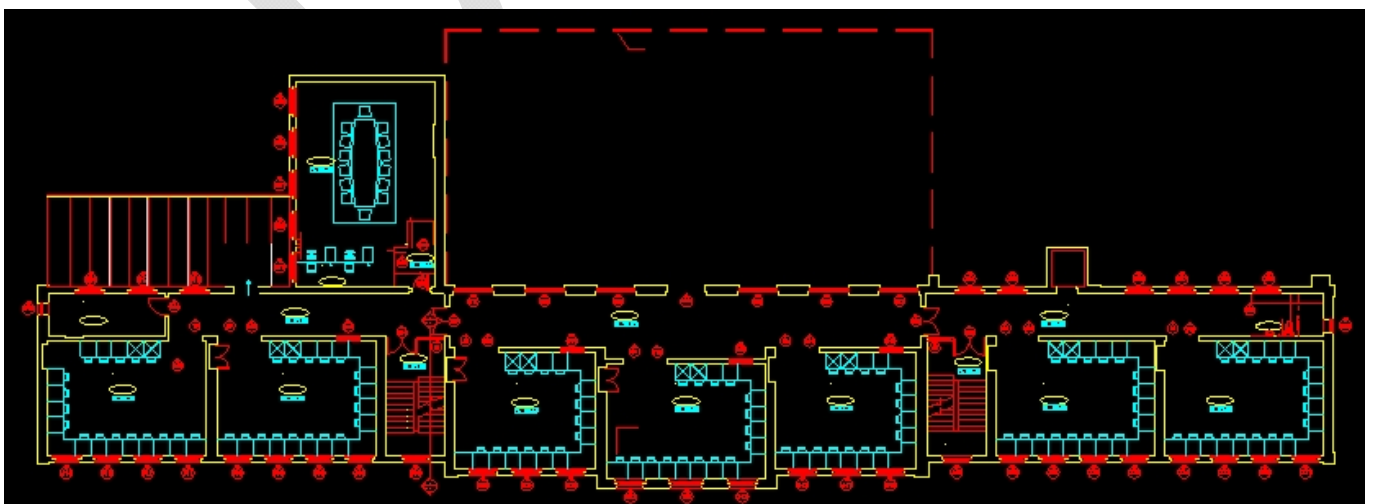


Figure 2: Kingsley Building – 1st Floor plan.

System Requirements (General):

- The source code file containing the **main()** method should be named **CBldgKingsley.java** and hence, the compiled byte code class file **CBldgKingsley.class**.
- Attempt to convert the resulting **JFrame** application into a **JApplet** and discuss any difficulties in achieving this. The source code file should be named **CBldgKingsleyApplet.java** and hence, the compiled byte code class file **CBldgKingsleyApplet.class**. Access to the **JApplet** should be via the *.html file **CBldgKingsleyApplet.html**.
- The application **must** be demonstrated (see below).
- Create a **JFrame** application, which opens to full screen resolution.
- Enable viewing as a full screen applications, where it is possible to suspend the windowing system (**JFrame** and the usual GUI components i.e. **JMenuBar**) so that drawing can be done directly to the screen.
- A **JMenuBar** with at least the **JMenu**'s of 'File', 'Navigate', 'Options' and 'Help'
- **JMenuItem** 's relating to **JMenu**'s e.g. 'Forward', 'Rotate' (Navigate), 'Floor 1', 'Floor 2', 'Floor 3', 'Clear' (Options) 'About item', 'Help' instructions (Help)
- Application icon of your choice.
- **JFrame** title set as **C2D3D GraphicsApp Application**.

System Requirements (3D):

- Create a **VirtualUniverse** (not a **SimpleUniverse**) to contain your Kingsley Building.
- Use the same thickness for internal walls and external walls and state this clearly in your report (together with any other dimensional assumptions).
- Use the same thickness for floors/ceilings and state this clearly in your report (together with any other dimensional assumptions).
- Walls, floor and ceiling should all 'look' different.
- Horizontal and vertical navigation (back, forward, left, right, up, down and rotations) within the maze/floors should be possible
- The following internal dimensions for the pair of end rooms is to be assumed: Width xxxxmm * Length xxxxmm * Height xxxxmm). Width applies to the Window wall.
- The Kingsley building floor plan **Figure 2** shows 7 rooms to the South East elevation and 1 room to the rearward facing North-West elevation. The two stairwells will fill the remaining width/space of the front elevation and corridor split the front 7 rooms from the 1 rear room. The floor plan layout can remain the same on all 3 floors (Clearly indicate **ALL** other dimensional assumptions in your report).
- Benching should be rendered in at least one computer room.
- Attempt to convert the **JFrame** application into a **JApplet** and discuss any difficulties in achieving this.

Additional functionality/complexity (3D) could include:

- Doors, windows, strip lights, tables, chairs and computers rendered in at least one room on the South East Elevation/Side of the Kingsley building (See **Figure 3**).
- Doors, windows, walls, roof rendered to the exterior of the Kingsley Building.
- The surfaces defined and rendered as a material or given a texture (digital camera images can be used).
- Incorporation of lighting effect using material, ambient, directional, point and spot lighting (to model windows and/or strip lights).
- User defined geometry (i.e. not primitive shapes). Preferred modelling package is

Blender (see resources below).

- Loaded geometry using object files e.g. additional items rendered within the rooms/building or outside.
- The external environment, floors, stairs, corridor and rooms should be navigable.
- Navigation using buttons and/or keyboard and/or mouse.
- Collision detection.
- Animation e.g. door/s opening, clouds moving past the windows.

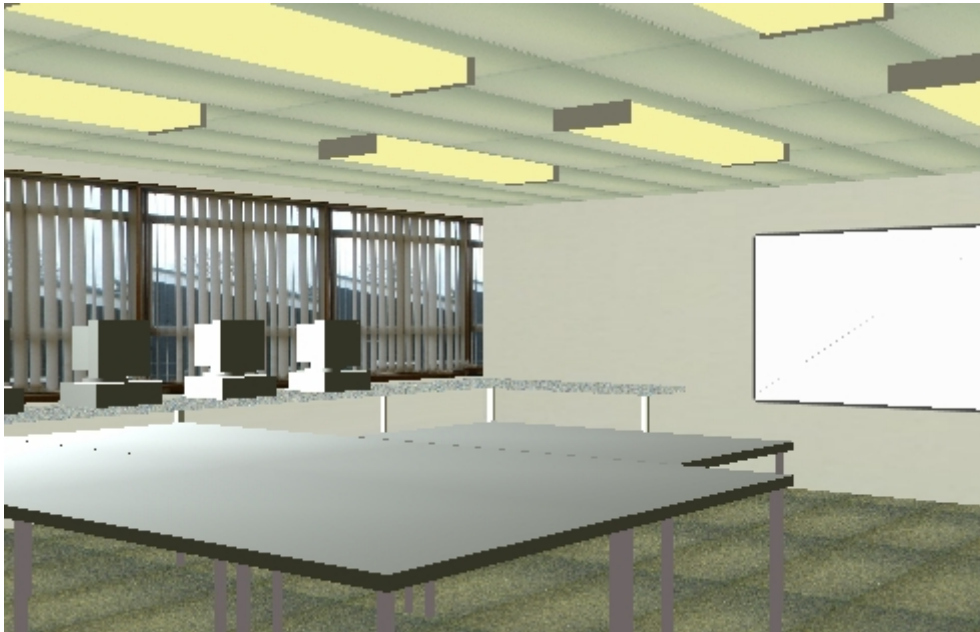


Figure 3: Kingsley – Java 3D View of Room showing rendered items.

Optional - Additional functionality/complexity (2D) could include:

- The first floor plan could be rendered into the application, this could be drawn in using Java 2D or imported from one of the supplied DXF drawing resources.

Grading:

A **bare pass** will involve displaying the Kingsley building, the first floor corridor and one room.

A **good pass** will include horizontal navigation within the scene and some of the 'Additional functionality/complexity' above.

A **very good pass** will incorporate most of the 'Additional functionality/complexity' above.

Technical Report

The report should be **5000** words and structured as follows:

- **Std. Front Sheet**
- **Title Page**
- **Table of contents**
- **Section A**
- **Introduction** / Problem statement for the task
- **Analysis** and assumptions for the task
- **Design** for the task in either pseudo-code or flowchart form (include GUI design fully labelled)
- **Implementation** (Explanation of the code – with screen shots).
- **Testing** (A test plan for the task – with screen shots)

- **Conclusion & Recommendations**
- **Section B**
- **Introduction** / Problem statement for the task
- **Analysis** and assumptions for the task
- **Design** for the task in either pseudo-code or flowchart form (include GUI design fully labelled)
- **Implementation** (Explanation of the code – with screen shots).
- **Testing** (A test plan for the task – with screen shots)
- **Conclusion & Recommendations**
- **References**
- **Bibliography**
- **Appendices**

In addition to the usual content, the appendices **must** include a commented file listing of **ALL** source code, **ALL** files on disk and an electronic version of the report in word (or similar).

It is expected that your report **should** make appropriate use of screen shots, particularly during the Implementation and Testing sections.

Viva/demonstration

The viva/demonstration of the developed application will be conducted within the timetabled session for **Week 21 (Tuesday 26/05/09)**. A timetable of times and names will be distributed the week before. This is a **compulsory** activity, which is considered an essential element of this assignment. The expected viva duration will be between 10 to 15 minutes. Present will be the Module Tutor and the student under assessment.

Resources:

Please refer to: **Guide To Writing Technical Reports** at <http://www.computing.northampton.ac.uk/~gary/BScC/reportwriting.html> for guidance on writing your report.

Freeware screenshot software MW Snap available from:
<http://www.mirekw.com/winfreeware/mwsnap.html>

Freeware DXF drawing viewer, freedwgviewer is a loader, viewer and measurer. Can load DXF files, turn layers on and off, can measure/determine dimensions, but cannot then save back as DXF. <http://www.infograph.com/products/dwgviewer/>

DXF Elevation & Plan Drawings will be available on the Computing Division file server, the direct location will be explained during class, but will be in /Turing/public/csym021/20082009.

author: Gary Hill
e-mail: gary.hill@northampton.ac.uk
Last modified: 03/02/09
© University of Northampton

MSc Computing (Internet Technology)

Java Programming (Assignment: 2D/3D Computer Graphics).			
Due for Issue <small>(week commencing):</small>	Tuesday, 03 February 2009	Date for Submission:	Friday, 30 May 2009
Agreed Date for late submission:		Module Tutor:	Gary Hill
Student Name:		Student ID	
Signature:			

Assessment Feedback

Aspect (& weighting)	Excellent 70-100% (D)	Very Good 60-69% (M)	Satisfactory 50-59% (P)	Needs some more work 45-49% (F+)	Needs much more work 0-44% (F-)
Introduction, Problem Specification (0.5):					
Analysis (Grasp of requirements) (1.5):					
Design (1.5):					
Implementation (Quality of Coding) (4):					
Testing Strategy & Testing (incl. Screen shots) (0.5):					
Ability to Develop Conclusions, Recognise Limitations of Work (1):					
Report Presentation (Format, Layout, Grammar, Syntax, Spelling) (0.5):					
Viva/Demonstration (0.5):					
Specific aspects of the assignment that the marker likes:			Specific aspects of the assignment that need more work:		
Tutor's Signature:			Date:		Grade:

The University of Northampton Policy on Plagiarism & Mitigating Circumstances will be strictly implemented. By submitting this assignment you are asserting that this submission is entirely your own/individual (or group – where

appropriate) work.

CSYM 021


Java Programming

2009

Assignment:

Development of 2D/3D Graphics Software: (100%)

Notes on achieving the most from this assignment and where marks can be lost:

- 1 Carefully read and read again the assignment brief. Following the instructions given carefully and check the front sheet to see where marks are allocated.
- 2 Always write in the third person i.e. **NO** I thought this, I did this etc. If you need to refer to yourself use "The author felt" etc.
- 3 The brief states that "The applications **must** be demonstrated". Anyone not demonstrating receives an F/F- even if the assignment is handed in.
- 4 If an assignment is not submitted (even if the application was demonstrated) a mark of G is given.
- 5 The brief states that "The source code file containing the **main()** method and the compiled byte code class files should be named as follows: **CBldgKingsley.java** & **CBldgKingsley.class**".
- 6 A "Std. Front Sheet (Attached to this assignment)" should be attached to the assignment with a Title Page.
- 7 The introduction/problem statement should repeat the information given in the assignment brief (available in  word).
- 8 Implementation should be a commentary of the code explaining how the design has been applied. Screen shots should shows snippets of the code and the outcome in the GUI.
- 9 The FULL source code in the Appendix should have a header indicating the usual information e.g.

```
/**
Program:    Assignment 2: Robot Path Finder Application
Filename:   RobotPathFinder.java
@author:    © Gary Hill (200WXYZ)
Course:     BSc/HND/HNC Computing Year 1
Module:     CSY1020 Problem Solving & Programming
Tutor:     Gary Hill
@version:   2.0 Incorporates Artificial Intelligence!
Date:       27/10/07
*/
```
- 10 Meaningful variable names should be used in code e.g. jBrotate (for the rotate JButton).
- 11 Your code should have meaningful comments e.g. `//button to rotate 90° clockwise.`
- 12 The aim of the Conclusion & Recommendations section is to demonstrate that you can develop your own conclusions and recognize the limitations of what you have produced. To achieve this you should clearly indicate:
 - the aim and objectives of the assignment from the assignment brief;
 - which objectives and items of general and advanced functionality have been met;
 - which have not and how you think they could have been solved given more time;
 - what are the limitations of what you have been asked to produce;
 - how would your approach differ given the opportunity to do the assignment again?

Division procedures for students 2008/9

*Except where specifically stated in the assignment brief, assignment work submissions must be word-processed with a footer comprising: your **name (family name underlined)**; **registration number**; **module code**; **date**; **page (M of N)**; and submitted in a single A4 plastic folder. The assessed work form should NOT be stapled to the coursework.*

Work submitted on disc (CD or DVD) must have your **name** electronically as the root directory name, and written on the disc itself in permanent ink: **name (family name underlined)**; **registration number**; **module code**; **year (e.g. 2007/8)**

*The Division cannot accept responsibility for lost assignments, so you must ensure that you keep a complete copy of the work you hand in ***.*

Submission of assignment work

The assignment work (including a completed Standard Front Sheet) is submitted to the Student Assessment Office at Avenue Campus. Opening hours are Monday to Friday - 10am - 4pm. After 4pm and before 7pm, all assignments and receipts can be put in the deposit box outside the Student Assignment Office on the MR corridor

(see <http://www2.northampton.ac.uk/academicregistry/ARHP/student-assignment-office>).

Anything in the drop-box at that time will be treated as if it had been handed in on the due date, UNLESS SPECIFICALLY INDICATED OTHERWISE ON THE ASSIGNMENT. If you have submitted work, it is your responsibility to remember to collect a copy of the receipt form from the relevant Student Assignment Office once the office re-opens. If a receipt is not available for collection you should contact your tutor and the School office immediately to give a copy of your work as defined in the previous section***.

Late work

Work that is handed in late must be submitted to the Student Assessment Office at Avenue Campus and work that is handed in outside of office hours must be submitted to the drop-box.

Penalties for late submission

Except where an extension of the hand-in deadline date has been approved (using the Request for Assignment Extension form), lateness penalties will be applied in accordance with University policy as follows:

(Working) Days Late	Penalty
1 - 7	maximum grade that can be achieved: P-
more than 7	G

Extenuating circumstances

If you believe that there are circumstances that justify an extension of the hand-in deadline for assignment work, you are required to use the Request for Assignment Extension form (available from the Programme Support Team Office MR82.) Extensions (to a maximum of 3 weeks in exceptional circumstances) are granted when there are serious and exceptional factors outside your control. Everyday occurrences such as colds and hay fever do not normally qualify for extensions. Where possible, requests for extensions should be made **before** the hand-in date.

The Division considers extenuating circumstances to be conditions that significantly impact on your work. Typically these will cover more than one module. Requests for consideration of mitigating circumstances in respect of assignment work submission, should be made using the mitigating circumstances form (see

http://www2.northampton.ac.uk/portal/page/portal/AcademicRegistry/ARHP/mc_contents). You are advised to speak to your Course Leader/personal tutor prior to completing these forms. Whilst mitigating circumstances are being considered, you are advised to inform relevant staff members, **and continue with the assignment**. Extensions of up to 3 weeks can be granted in order that you can submit the current piece of assessed work. If the circumstances are too serious to be resolved by such an extension, alternative arrangements involving a different piece of work may be made.

Feedback

Feedback (which may be generic or oral) will be given to the class within 15 working days of the assignment hand-in date. This may be done in the first 15 minutes of a lecture. This will be followed by individual written feedback tying to the Learning Outcomes listed in the assignment brief, together with any additional helpful feedback such as areas of strength and areas for improvement.

Contacting Staff

All academic staff are available to students outside of their class contact time. Availability may be indicated in a number of ways: marked outside their office doors (along with an appointments sheet); at the end of their emails; marked on their timetables. Most staff would prefer you to contact them by email in the first instance.

When emailing staff please ensure that you indicate in the **subject field** the purpose of your email (including **the module number**), e.g. *CSYM010 project appointment request; CSYM020 query on assignment hand in using button wallet rather than A4 folder*. Since staff receive several hundreds of emails daily with external emails sent to junk mail, it is important that staff can locate those emails which are not junk.

Cheating

The consequences of cheating in assessments are serious - you will fail the module. Cheating is using or attempting to use unfair means to enhance performance. This includes plagiarism (presenting someone else's work as if it was your own), collusion (working with others on an individual assignment), taking prohibited material into examinations and allowing other students to access your work. Make sure that you do not give someone the opportunity to steal your work (e.g. *by asking them to print it out for you*). We tell students about cheating both during induction and in your student handbook, but if you have any doubt about what cheating is or how to reference material properly, please ask a tutor. We recommend that you use the Harvard system for referencing.

The assignment Standard Front Sheet used for each assignment requires you to sign to confirm that the work submitted is your own work and that any information and material used has been properly identified and acknowledged. The University operates an electronic plagiarism detection service where your

work may be uploaded, stored and cross-referenced against other material. The software searches the World Wide Web and extensive databases of reference material to identify duplication.

For detailed information on the procedures relating to plagiarism, please see the current version of the Academic Registry guide on Academic Integrity, see: <http://www2.northampton.ac.uk/portal/page/portal/AcademicRegistry/ARHP/Guides%20to%20Procedures>)

Reassessment and Revision

Reassessment in written examinations and coursework is at the discretion of the Course Assessment Board and is dealt with strictly in accordance with University policy and procedures.