# 12ATAR Computer Science
# Project 1 – Software Development

## Part 1

| | |
|---|---|
| Time Allocation: | 3.5 weeks |
| Due Date: | Friday 5th April by 9:00pm (Term 1 Week 9) |

## Part 2

| | |
|---|---|
| Time Allocation: | 4 weeks |
| Due Date: | Friday 17th May by 9:00pm (Term 2 Week 3) |

# Introduction

Before Fortnite, games were much simpler! In fact, before graphics cards came along most games were text-based, with no graphics at all!! One of the most common types of games were text-based role-playing games (RPG) where the player controls the actions of a character as they move through an imaginary world.

In this project, you will use the Software Development Cycle (SDC) as part of the System Development Life Cycle (SDLC) to develop your own interactive text-based RPG. Your game will allow a player to move their character through their world, picking up items, fighting battles and/or solving puzzles along the way.

For more information on role playing games, and to look at some examples, have a look at the following Wikipedia pages:
> https://en.wikipedia.org/wiki/Role-playing_video_game
> https://en.wikipedia.org/wiki/The_Hobbit_(1982_video_game)
> https://en.wikipedia.org/wiki/Zork
> https://en.wikipedia.org/wiki/Rogue_(video_game)

To successfully develop your RPG, you will need to make use of:
- Simple and complex data types, including arrays and records
- A variety of control structures, including selection and iteration
- Modularisation, including parameter passing and the use of functions

You can make your game purely text-based or you could create a more visual interface – the choice is up to you. Whatever option you choose, it must be intuitive for the user to play and allow the user to easily move around the world and carry out tasks.

If you are unsure if your idea for a RPG will meet the requirements, then make sure that you consult your teacher well before the project is due!

# Task Requirements

## Part 1

### Preliminary Analysis
- Create a Gantt Chart to outline your project plan. You should break the task down into the steps that you need to complete in order to produce a suitable system. This will enable you to gain an understanding of what you need to do and also give you a timeline to complete your task.

### Analysis
- Provide a detailed description of your RPG. This should include:
  - a synopsis of the game, including the background story to the game
  - a description of the main characters that will appear in the game, including any powers or special abilities they may possess
  - a description of how the various characters might interact
  - a description of how the objectives of the game and how they might be reached
- Discuss any constraints and limitations that you have identified. As part of this discussion you should consider:
  - the potential users of the game
  - how users will play the game
  - input and output requirements
  - development requirements, including:
    - methodology to be used
    - time constraints
    - technical constraints (including development skills required)
  - data storage and security issues
  - any legal and/or ethical issues that need to be considered with regards to the development of the game
- Produce a detailed list of requirements for your RPG.

### Design
- Create a map of your world, showing the main features that the main character will need to navigate through
- Using the Software Development Cycle (SDC), design a software solution for the system.
  - Define the structure of your program by creating a structure chart to show how the various modules will interact
  - Define the interactions that the main character will be able to have with the other characters and objects in the game
  - Develop a chart showing the commands that can be used and what those commands will do
  - Use pseudocode to design any data structures that you will need
  - Use pseudocode to design an algorithm to show:
    - How the main character will move through the world
    - How the main character will interact with other characters in the game (NOTE: you should provide some randomness to these interactions so the game is not always the same)
  - desk check your algorithms to ensure there are no logic errors
  - produce a mock-up of the proposed user interface for you RPG
  - discuss your user interface design in terms of how it meets the user needs and considers the factors that you identified in the analysis stage

## Part 2

### Development

- Code your program using C#. You can either create a console application using entirely text-based interface or a WPF application to create an interface using a mixture of images, text, buttons and other controls.
  Your code should include extensive internal documentation, including comprehensive comments throughout your code, describing the purpose of each module and explaining any complex sections of code. Part of your internal documentation should include the use of appropriate naming conventions
- Develop a detailed test plan for your program to ensure that it is fully tested (include any necessary test data)
- Produce relevant user documentation to explain to the user how to use the system. This should be included as part of the program itself so that the user can easily get help while playing the game.

### Evaluation

- Reflect on the success of your system and how well it meets the system requirements. To perform your evaluation, you should:
  - Consider how well your program meets the system requirements you developed in Part 1
  - Consider the structure chart, data structures and algorithms you developed in Part 1 and compare them to your finished program. Discuss any differences and explain why you have made these changes.
  - Consider what aspects of the game could be improved and the quality of the user experience
  - Document any known bugs in the game and explain how they impact the performance of the game
- Document the sources you have used to get information about how to develop your system, including all websites and textbooks.

# Submission

This task has been divided into 2 sections:

> **Part 1** is due on **Friday 5th April** by **9:00pm** (Term 1 Week 9)
> **Part 2** is due on **Friday 17th May** by **9:00pm** (Term 2 Week 3)

## Submission Requirements

### Part 1
For Part 1, you are to produce a single, well formatted PDF document. This document should include:
- A cover page
- Suitable headings to make each section clear
- Headers and footers
- Appropriate terminology, explanations and written expression
- Diagrams created using appropriate software

This file should be named **YourSurname_Project1Part1** and should be submitted via **Nexus**.

### Part 2
For Part 2, you are to produce:
- a single, well formatted PDF document that includes all changes to the design and your evaluation. This document should be named **Project2Part2Documentation** and should include:
  - A cover page
  - Suitable headings to make each section clear
  - Headers and footers
  - Appropriate terminology, explanations and written expression
  - Diagrams created using appropriate software (if appropriate)
- a single folder that should contain the entire project for our application. You should ensure that any necessary libraries, images and/or sound files are included as part of your project. Compress the project folder to a **zip** file and name the zip file **Project2Program**.

You should submit both these files to **Nexus**

Remember, appropriate naming conventions and appropriate presentation are essential skills for any software development project, so **work that is not named correctly or submitted in the correct manner will have marks deducted**. If you have any questions regarding how to submit your work, make sure that you check with your teacher exactly what is required!