# Forest Camp Scene

Ross Chapman
40209091@napier.ac.uk
Edinburgh Napier University - Module Title (SET08116)

## Abstract

The goal of this project is too create a decent looking forest camping scene. Some features that are implemented are a free camera, texturing on different geometry shapes and transforming those in order to get them in the right location and orientation. Some features that will be implemented are point and spot lights, more cameras,

**Keywords –** Graphics, Napier

## 1 Introduction

The motivation for this project is the forest and trying to emulate that with a camping scene.

Some effects/functions that were used are: .Free Camera so that there can be any number of different viewpoints that can be achieved. .Loading in textures from downloaded sources. .Using transforms to correctly place my geometry.

**Referencing** You should cite References like this: [1]. The references are saved in an external .bib file, and will automatically be added ot the bibliography at the end once cited.

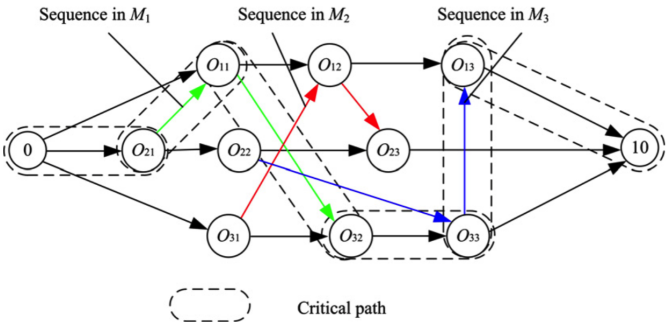


Figure 1: **ImageTitle** - Some Descriptive Text

## 2 Formatting

Some common formatting you may need uses these commands for **Bold Text**, *Italics*, and underlined.

### 2.1 LineBreaks

Here is a line

Here is a line followed by a double line break. This line is only one line break down from the above, Notice that latex can ignore this

We can force a break
with the break operator.

### 2.2 Maths

Embedding Maths is Latex's bread and butter

$$J = \left[\frac{\delta e}{\delta \theta_0}\frac{\delta e}{\delta \theta_1}\frac{\delta e}{\delta \theta_2}\right] = e_{current} - e_{target}$$

### 2.3 Code Listing

You can load segments of code from a file, or embed them directly.

Listing 1: Hello World! in c++

```
1 #include <iostream>
2
3 int main() {
4     std::cout << "Hello World!" << std::endl;
5     std::cin.get();
6     return 0;
7 }
```

Listing 2: Hello World! in python script

```
1 print "Hello World!"
```

### 2.4 PseudoCode

**for** $i = 0$ **to** $100$ **do**
    print_number = true;
    **if** *i is divisible by 3* **then**
        print "Fizz";
        print_number = false;
    **end**
    **if** *i is divisible by 5* **then**
        print "Buzz";
        print_number = false;
    **end**
    **if** *print_number* **then**
        print i;
    **end**
    print a newline;
**end**

**Algorithm 1:** FizzBuzz

# 3 Conclusion

# References

[1] S. Keshav, “How to read a paper,” *SIGCOMM Comput. Commun. Rev.*, vol. 37, pp. 83–84, July 2007.