A Welsh *Hello World* Program

Sebastian Danicic
Supervisor: Professor Heinrich Wormold

January 29, 2011

Part I

The Report

Abstract

This project is the first to produce a ‘Hello World’ program in Welsh.

Acknowledgements

I am hugely grateful to the people who helped me. Thank you darling mother for giving birth to me. Without you and daddy this project would not have been possible.

I would also like to thank Professor Heinrich Wormold for his expert tuition in Java and Welsh.

I would like to thank my Cat, Dennis, for help with Java programming and to last but not least to the Merlin the Wizard for leading me to a golden age of peace, righteousness, and prosperity.

Contents

List of Figures

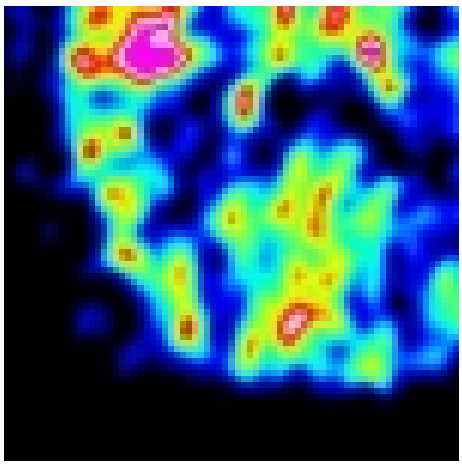
List of Tables

Chapter 1

Introduction

1.1 The Hello World Problem

In this project I am attempting to solve the well-know hello world problem. In this problem we tried to make the computer display **hello World** on the screen using the Object oriented programming language, Java. Although there have been many solutions in other languages including Mandarin Kuhn et al. (2008), we believe this is the first solution to the problem in Welsh.



1.2 Why my problem is interesting

This problem is very interesting as there have been many proposed solutions all of which have their drawbacks. These are discussed further in In Chapter 2 on page 3.

1.3 The layout of my report

You probably write this section last (but before the abstract).

In Chapter 2 on page 3 I discuss the background to my Project.

In Chapter 6, Page 3 I evaluate my project. In Section 6.2 I discuss how well I met the original goals of my project.

Chapter 2

Background

2.1 The Problem being solved

The aim of the project is to produce a Welsh version of the hello world program. In the process we hope to forge links between digital technologies and artistic practice. We will develop original theories, new departures in the technological arts and pioneering, and commercially viable, to produce a sustainable software product. In order to complete the project I will undertake research into work from the culture industries, film production, design consultancies, museums and galleries, interactive media content providers, software development houses, and hardware design laboratories - to explore and extend the use of today's digital technology and to define and implement the art and design technologies of tomorrow.

In Figure 2.1 page 4 is a pretty picture.

2.1.1 How other People Have Solved it

Hello World in C

In Section 2.1.1 I talk about how other people have so

The first known instance of the usage of the words "hello" and "world" together in computer literature occurred earlier, in Kernighan's 1972 Tutorial Introduction to the Language Kuhn et al. (2008), with the following code:

```
main( ) {
  extrn a, b, c;
  putchar(a); putchar(b); putchar(c); putchar('!*n');
}
a 'hell';
b 'o, w';
c 'orld';
```

Apart from not being in Welsh, its obtuse structure clearly fails to forge links between digital technologies and artistic practice.

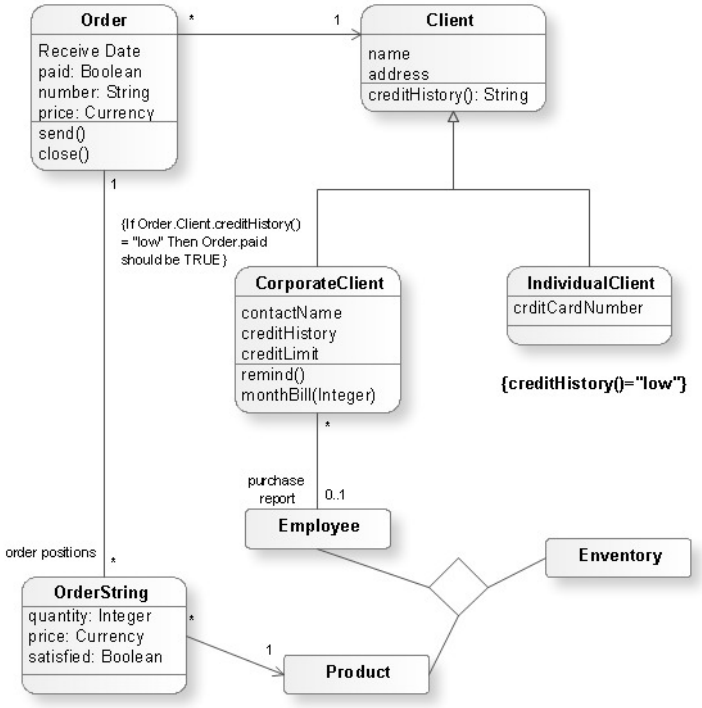


Figure 2.1: Pretty Picture

(Part of the learning objectives of level 3 (which is where you are now) is the exercise of critical judgement; it is up to you to judge critically what appropriate titles and wordings for your report are, based on academic reports you have read during your course of study.)

Hello World in Erlang

In 1805 Gottfried Llantwit-Major Author and Co-author (1352) produced the first Erlang version:

```
-module(hello).

-export([hello/0]).

hello() ->
   io:format("Sut dach chi?", []).
```

Although this is very good attempt, as the reader will observe, it produces Welsh output. Unfortunately however, it does not translate correctly to Hello World.

A more Goldsmithian Example

```
#include <stdio.h>
    main(){int i,n[]={(((1<<1)<<(1<<1)<<(1<<
    1)<<(1<<(1>>1)))+((1<<1)<<(1<<1))), (((1
    <<1)<<(1<<1)<<(1<<1)<<(1<<1))-((1<<1)<<(
    1<<1)<<(1<<1))+((1<<1)<<(1<<(1>>1)))+ (1
    <<(1>>1))),(((1<<1)<<(1<<1)<<(1<<1)<< (1
    <<1))-((1<<1)<<(1<<1)<<(1<<(1>>1)))- ((1
    <<1)<<(1<<(1>>1)))),(((1<<1)<<(1<<1)<<(1
    <<1)<<(1<<1))-((1<<1)<<(1<<1)<<(1<<(1>>1
    )))-((1<<1)<<(1<<(1>>1)))),(((1<<1)<< (1
    <<1)<<(1<<1)<<(1<<1))-((1<<1)<<(1<<1)<<(
    1<<(1>>1)))-(1<<(1>>1))),(((1<<1)<<(1<<1
    )<<(1<<1))+((1<<1)<<(1<<1)<<(1<<(1>>1)))
    -((1<<1)<<(1<<(1>>1)))),((1<<1)<< (1<<1)
    <<(1<<1)),(((1<<1)<<(1<<1)<<(1<<1)<<(1<<
    1))-((1<<1)<<(1<<1))-(1<<(1>>1))),(((1<<
    1)<<(1<<1)<<(1<<1)<<(1<<1))-((1<<1)<< (1
    <<1)<<(1<<(1>>1)))-(1<<(1>>1))), (((1<<1
    )<<(1<<1)<<(1<<1)<<(1<<1))- ((1<<1)<< (1
    <<1)<<(1<<(1>>1)))+(1<<1)), (((1<<1)<< (
    1<<1)<<(1<<1)<< (1<<1))-((1<<1)<< (1<<1)
    <<(1<<(1>>1)))-((1<<1) <<(1<< (1>>1)))),
    (((1<<1)<< (1<<1)<<(1<<1)<< (1<<1))- ((1
    <<1)<<(1<<1)<<(1<<1))+((1<<1)<< (1<<(1>>
```

```
1)))), (((1<<1)<<(1<<1) <<(1<<1))+(1<<(1
>>1))),(((1<<1)<<(1<<1))+((1<<1)<< (1<<(
1>>1))) + (1<< (1>>1)))}; for(i=(1>>1);i
<(((1<<1) <<(1<<1))+((1 <<1)<< (1<<(1>>1
))) + (1<<1)); i++) printf("%c\n",n[i]); }
```

This solution typically involves a group of participants standing or sitting together in a circle, with a side-lying, empty bottle on the ground at their epicentre. The bottle functions like the spinner for a standard board game, with its opening serving as the pointer. Here we explore alternative or additional overlays that can be applied including bin-the-spottle, spin-the-bottle, hi-fi, show you mine if you show me yours, seven minutes in heaven, hopes, fears and dreams and sight for sore eyes.

2.2 The Science and Technology needed to solve it

2.2.1 In various sections, describing the technological tools you'll use

We conduct an exploration of analog and digital bodies, using a range of technologies to remap the solid and obdurate real of bodies into the dispersions and virtualities of the digital, and then back again into real physical spaces. The "Hello World" of the title means at least three things.

Firstly, the pasting of viewpoints together, the suturing of the subject into the avatar. Secondly, paste as glue, as half-liquid and half solid, as a materiality of renewable and infinite pliability. This is the chora of the avatar, the body matrix that is less a framework than a smearing of paste. And thirdly, paste as pasty and dis/comfortable substance, paste as slimy and dripping. While this abjection is already implicit in paste as glue, the pastiness of paste involves the projection and dreaming through of the avatar, the inhabitation of avatar bodies and the emptying of real bodies into the avatar.

"Hello World" comes out in avatar motions and behaviours. Firstly, these are formed by symbolic orders, presenting surfaces to read in terms of sexuality, power, emotion, and other projections. At the same time, the pasty avatar body tends towards collapse and abjection. Work on the avatar becomes a choreography of exposure and rupture, modelling and presenting inconceivable and untenable data, within which tensions and relationships are immediate and intimate. One might imagine, then, this inconceivable data as a form of organism itself: as part of a natural world or a world already given; out of this we might think through new ideas of landscape, wilderness, hard ecology, the earth itself.

The project will theorise and demonstrate these topics. The first part discusses theoretical frameworks. We will introduce the topic of dismemberment and telepresence in terms of the presence or appearance of abjection in Second Life avatars. He will connect this to the epistemology of emptiness vis-a-vis sheave theory and Buddhist philosophy, and then to the problems of motion and behaviour of avatars. Sandy Baldwin will discuss the topography of limits in Second Life, both body limits and spatial limits, an connect this to issues of the hunt and animal display.

We will also discuss the dynamics of performance and audience in Second Life. The second part of the project will show off Sondheim and Baldwin's approach to re-mapping live bodies into Second Life performances, including: video and other examples of motion capture and scanning; intermediate processing of files (e.g. editing .bvh data or working with Blender); and then the resulting works, including documents of Second Life performances and re-mappings back into "first life" spaces with dancers and other live performers. The final part of the project will include avatar performance by Sondheim, Baldwin, and other participants in Second Life.

Chapter 3

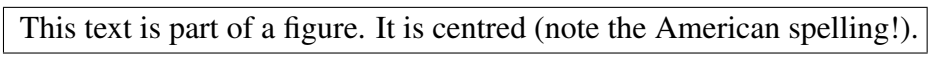
Project Description

3.1 Introduction

3.2 Textual Description of what you have built

3.2.1 Divided into subsections

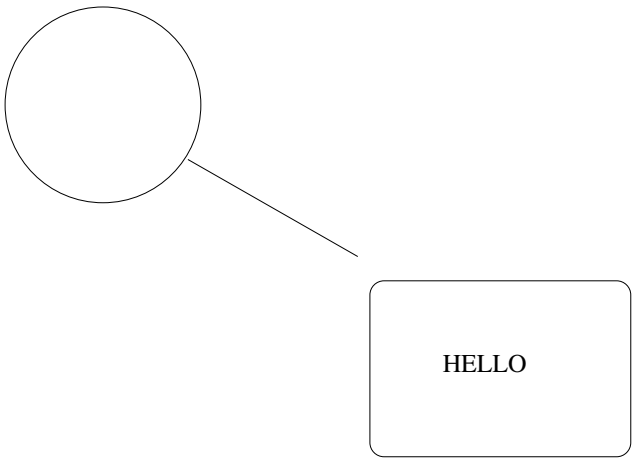
. . . for easy understanding.



Latex will put the figure at the place most appropriate for it,
according to publishing convention, automatically.

Figure 3.1: This is how you make a figure in your document.

3.2.2 Block Diagram



3.3 Work Plan

Describe the work plan you proposed in your second deliverable and assess whether you stuck to it.

Chapter 4

System Evaluation

4.1 How my system/software can be tested

4.2 The results of testing my system/software

4.3 How I might improve my project based on this evaluation

Chapter 5

Summary and Conclusions

5.1 Summary of the Project

5.1.1 Goals

The project aimed to...

5.1.2 Outcomes

The project achieved...

5.2 Conclusions Drawn from the Project

This was a fabulous/good/bad/terrible way to address this task. Here's why...

Part 1	Part 2	Part 3
Result A	Result B	Total

Table 5.1: This is how you make a table in your document.

Chapter 6

Project Self-Evaluation

6.1 Introduction

In this chapter, you reflect on what you did in your project and how well you did it.

6.2 Project outcomes

Did you meet the outcomes you said you would? If not, why not? How could you have done better?

6.3 Project planning

Did your planning work well? How could you improve it?

6.4 Self-evaluation

What you learned about yourself in doing this project.

Part II

appendix

Chapter 7

Java Program

```
class helloworld
{
        public static void main(String args[])
        {

        }

}
```

7.1 Useful Information

7.1.1 Help with LaTeX

You can get help with LaTeXand BibTex from http://www.math.harvard.edu/texman/ Kuhn et al. (2008), and from `http://en.wikibooks.org/wiki/LaTeX`.

> Author et al. (1927) argue that nothing is true. You use this if you need to make the authors part of your sentence. Look at the `.tex` file to see how this is done.

The second is as follows:

> It has been argued that nothing is true (Author et al., 1927). You use this if you don't mention the authors in your sentence. Look at the `.tex` file to see how this is done.

When you do this, you must have a definition for each item in your `DoCReport.bib` file. There is an example to help you, for journal articles (Author et al., 1927), books (Author and Co-author, 1352), chapters in collection books (Author, 2009), and miscellaneous things like web sites (Bloke, 1904). You can make up the keys for these things (that is, `ArticleA`, *etc.*); you just need to make sure that each is unique. It doesn't matter what order to place the entries in the `.bib` file.

If you compare the entries in the `.bib` file, you'll see that each is rendered differently into a bibliography entry at the back of the formatted report. LaTeXand BibTeX handle this correctly on your behalf – but make sure you read error messages and act on them, because the software can't correct any errors you make!

```
\begin{quote} \citet{ArticleA} argue that nothing is true. You use this i
need to make the authors part of your sentence. Look at the {\tt .tex} fi
see how this is done.
\end{quote}

The second is as follows:
\begin{quote}

It has been argued that nothing is true \citep{ArticleA}. You use this if
don't mention the authors in your sentence. Look at the {\tt .tex} file t
how this is done.

\end{quote}

When you do this, you must have a definition for each item in your {\tt
DoCReport.bib} file. There is an example to help you, for journal article
\citep{ArticleA}, books \citep{BookB}, chapters in collection books
\citep{CollectedC}, and miscellaneous things like web sites \citep{MiscD}
can make up the keys for these things (that is, {\tt ArticleA}, {\it etc.
just need to make sure that each is unique. It doesn't matter what order
place the entries in the {\tt .bib} file.

If you compare the entries in the {\tt .bib} file, you'll see that each i
rendered differently into a bibliography entry at the back of the formatt
report. \LaTeX and BibTeX handle this correctly on your behalf -- but mak
you read error messages and act on them, because the software can't corre
errors you make!
```

7.1.2 An important thing to note!!

LaTeX deals carefully with all the forward and backward referencing in your document (*e.g.,* tables of contents, *etc.*). This means that you sometimes need to run it as many as three times before all the inter-dependencies are sorted out. Make sure you read the output of the program, which will tell you when you need to run it again.

Bibliography

Author, A., Author, A. S., and Author, A. T. (1927). The title of the paper. *The name of the journal*, 1(1):1–100.

Author, A. D. and Co-author, A. (1352). *The book they wrote*. London University Press, London, UK.

Author, J. O. (2009). Just one chapter of many. In Editor, O. and Editor, A., editors, *The Collection of chapters*. Another University Press, Somewhere, New York.

Bloke, S. (1904). My web site. `www.foo.bar`; Last accessed: 10/10/1904.

Kuhn, R., Scott, R., and Andreev, L. (2008). An introduction to using TEX in the Harvard Mathematics Department. `http://www.math.harvard.edu/texman/texman.html`; Last accessed: 18/01/2008.