Computer Science 3B
Practical Assignment 00
2017-07-27

Deadline - 2017-07-27 17h00   **Marks: 45**


This practical assignment must be uploaded to eve.uj.ac.za **before** 2017-07-27 17h00. Late or incorrect submissions **will not be accepted**, and will therefore not be marked. You are **not allowed to collaborate** with any other student.

Good coding practices include a proper coding convention and a good use of commenting. Marks will be deducted if these are not present. See the reminder page for more details.

Artificial neural networks (ANN) are a computational model based on biological neural networks. The most basic component of a ANN is a **neuron**. Currently we do not possess the necessary instructions to creature a full ANN but we can start with the basics.

The most basic ANN is a network which contains a single node. This single node will take inputs and created a weighted sum with a bias to that weighted sum, and finally pass that output through an activation function. The output of the activation function becomes the output of the node. This single node is known as a perceptron and the figure below shows an example.
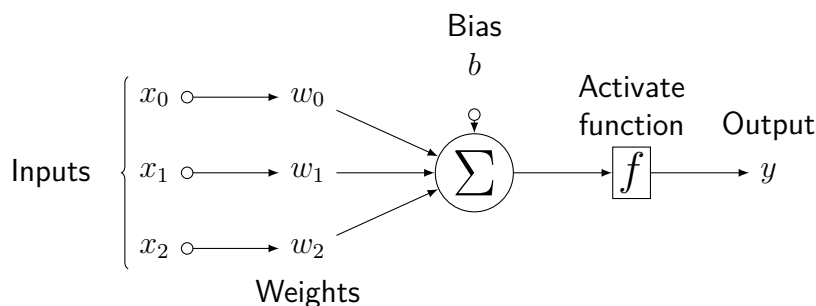
The focus of this practical is to create a perceptron. The weighted sum is calculated as follows:

$$s = x_0 * w_0 + x_1 * w_1 + x_2 * w_2 + b$$

where
s - weighted sum $x_{0...2}$ - input $0...2$ $w_{0...2}$ - weight $0...2$ b - bias

Since we cannot use branching instructions (if statements, loops, etc…) the activation function will be a simple division.

$$y = s/a$$

where
y - output s - weighted sum a - activation variable

Write an 80x86 assembly program that will calculate the output of a single perceptron. You are only allowed to create global variables for the terns specified above (i.e. no temporary variables). **Note**: all values are integers!

Your submission should include the assembly code that you wrote and pseudocode for your solution (alternatively c++ or java code).

Test data:

x_0	1	w_0	5	b	5
x_1	4	w_1	3	a	10
x_2	8	w_2	2	s, y	?

Mark sheet

1. Pseudocode   [05]

2. Weighted sum with bias   [05]

3. Activation function   [05]

4. Structure and layout (no temporary variables, correct data types)   [05]

5. Commenting   [05]

6. Correct execution.   [20]

NB

Submissions which **do not assemble** will be capped at 40%!

Execution marks are awarded for a correctly functioning application and not for having some related code.

Reminder

Your submission must follow the naming convention as set out in the general learning guide. **Example**

Surname	Berners-Lee	Initials	TJ
Student number	209912345	Module Code	CSC3B10
Current Year	2017	Practical number	P00

`Berners-Lee_TJ_209912345_ CSC3B10_2017_P00`

Your submission must include the following in a single zip (compressed) file:

- *Source file* (`asm` file) - File containing your solution. Your details must be included at the top of the source code.
- *Program design* (`pdf` file) - File containing your design. Your details must be included at the top of the design.