Name: _____ Student number: _____

Introduction to Scientific Computing
PSCB57
Midterm Exam

Professor Hanno Rein
University of Toronto, Scarborough

Monday, 20th October 2014

	Points	Max Points
Question 1		5
Question 2		5
Question 3		4
Question 4		5
Question 5		6
Bonus Question		5 (Bonus)
Total		25

- Duration: 60 minutes.
- No aid sheets, books or other notes are allowed.
- All electronic devices must be stored together with your belongings at the back of the room.
- Write your answers on the question sheet. If you need more paper raise your hand.
- The length of the white space for each question gives you an idea of the expected length and complexity of the answer.

Question 1 **5 Points**

The following appears on the screen of the linux machine that we have been using for the assignments.

```
hanno@rein001:~$ cd PSCB57
hanno@rein001:~/PSCB57$ cd assignment_04
hanno@rein001:~/PSCB57/assignment_04$ ls
INSTRUCTIONS.txt
hanno@rein001:~/PSCB57/assignment_04$ cat INSTRUCTIONS.txt
hanno@rein001:~/PSCB57/assignment_04$ vi fibonacci.py
```

Describe in a few words what each of the five commands does.

`cd PSCB57` _____

`cd assignment_04` _____

`ls` _____

`cat INSTRUCTIONS.txt` _____

`vi fibonacci.py` _____

Question 2 **5 Points**

We work in IEE754 double floating point precision. Circle all numbers that can be represented exactly.

1.0	2.0	0.0	-5.0	-1e+1
0.1	-0.2	1e+512	0.5	-0.0625

Question 3

4 Points

We still work in IEE754 double floating point precision. Calculate the following expressions.

`2 + 2 =` _____

`1e+24 + 1 =` _____

`(1e+67 + 1) - 1e+67 =` _____

`(((((0.25 + 1e-18) + 1e-18) + 1e-18) + 1e-18) + 1e-18) =` _____

Question 4

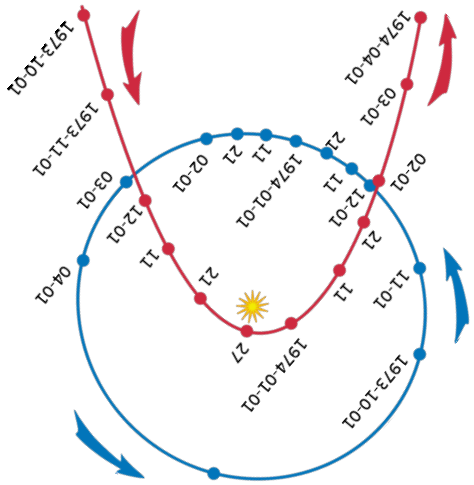
5 Points

```
SET 0 r9
SET 1 r7
SET 4 r1
SET -3 r3
PRINT r9
ADD r9 r7 r9
IF r9 r7
JUMP r3
```

What is the output of this assembler program? The syntax of assembler commands can be found on the last page.

Question 5

6 Points



The image shows the paths of the comet Kohoutek and the Earth. The comet is on a nearly parabolic orbit. If the coordinate system is chosen appropriately (you don't have to worry about this), its path can be described as a quadratic function:

$$f(x) = a_0 + a_1 \cdot x + a_2 \cdot x^2.$$

Astronomers have measured the position of the comet 4 times and give you their measurements as the following pairs: $(x_0, y_0), (x_1, y_1), (x_2, y_2), (x_3, y_3)$. Your task is to fit the function $f(x)$ and find the coefficient a_0, a_1 and a_2 using a linear least square fit. Let's start by writing down the system of equations describing the problem. Fill in the missing parts:

$$\begin{pmatrix} \_\_ \\ \_\_ \\ \_\_ \\ \_\_ \end{pmatrix} = \underbrace{\begin{pmatrix} \_\_ & \_\_ & \_\_ \\ \_\_ & \_\_ & \_\_ \\ \_\_ & \_\_ & \_\_ \\ \_\_ & \_\_ & \_\_ \end{pmatrix}}_{\equiv C} \cdot \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} + \begin{pmatrix} e_0 \\ e_1 \\ e_2 \\ e_3 \end{pmatrix}$$

The e_i values in the above equation are the residuals (errors). In the lecture, we minimized the e_is to arrive at the following linear system of equations

$$b = A \cdot a$$

where we introduced the matrix $A = C^T \cdot C$ and the vector $b = C^T \cdot y$. Describe, qualitatively in equations or words, how you would solve the system of linear equations $b = A \cdot a$ for a using the LU decomposition.

Bonus Question

5 Bonus Points

In the problem from the previous question calculate a_0, a_1 and a_2 given the data points $(-2, 3)$, $(-1, 1)$, $(1, 1)$ and $(2, 2)$. Note that you can solve the linear system of equations without the LU decomposition for these specific numbers.

Appendix A Assembler commands

- ADD r1 r2 r3
  This command adds the value stored in register r1 to the value stored in r2 and stores the result in r3.
- IF r1 r2
  This command only executes the next command if the value in register r1 is bigger than the value in register r2.
- COPY r1 r2
  This command copies the value in register r1 to register r2.
- SET s1 r1
  This command sets the value in register r1 to the number s1 (s1 is just a number, not the address of a register).
- JUMP r1
  This command moves the instruction pointer by the value of r1.
- PRINT r1
  This command prints the value in register r1 on the screen.