SECTION B

NUMERICAL METHODS 1 (3.09) taught by David Ham

Candidates being examined in "Numerical Methods 1" should answer at least one question from section B.

B1. (i) Convert the numbers in the following problems into 4 bit two's compliment signed binary integers before performing the calculation and converting back to base 10:

(a) $5 + 2$

(4 marks)

(b) -2×2

(5 marks)

(ii) For each of the following Python functions, write a mathematical expression using only matrices and vectors which performs the same operation. In each case, state whether each input and output is a matrix or vector.

(a)
```
def function_1(a,b):
    from numpy import dot, zeros

    c=zeros((a.shape[0],b.shape[1]))

    for i in range(a.shape[0]):
        for j in range(b.shape[1]):
            c[i,j]=dot(a[i,:],b[:,j])

    return c
```

(4 marks)

(b)
```
def function_2(a,b):
    from numpy import dot, zeros

    c=zeros(a.shape[1])

    for i in range(a.shape[1]):
        c[i]=dot(a[:,i],b)

    return c
```

(4 marks)

(iii) The forward difference approximation to the derivative of a function f at a point x is given by:

$$f'(x) = \frac{f(x+h) - f(x)}{h} + \mathcal{O}(h^p)$$

for some p. By expanding the first two terms of the Taylor series for f, derive this formula and find the value of p.

(8 marks)

SECTION CONTINUED ON NEXT PAGE

B2. (i) Consider the number -2.5

(a) Write the number in the form:

$$s1.m \times 2^{e-b} \tag{1}$$

Use a floating point format with 3 exponent bits and 4 mantissa bits, and a bias of 3. All the numbers must be written in binary.

(6 marks)

(b) Convert this number into the bit pattern which would actually be stored, according to the layout provided on the formula sheet.

(2 marks)

(ii) Consider the following piece of Python code:

```
def find_root(f, a, b, eps):
    if f(a)*f(b) > 0:
        return None, 0

    while b-a > eps:
        m = (a + b)/2.0
        print a,b,b-a,m
        if f(a)*f(m) <= 0:
            b = m  # root is in left half of [a,b]
            print 'Root in left half'
        else:
            a = m  # root is in right half of [a,b]
            print 'Root in right half'
    print m
    return m
```

(a) Which of the root finding algorithms which we studied in the course does this function implement?

(2 marks)

(b) Imagine I run the following Python code:

```
def g(x):
    return x

find_root(g, -0.5, 1., 0.5)
```

What is printed out?

(7 marks)

(iii) Suppose we have an $n \times n$ matrix A and three n-vectors $\mathbf{b}$, $\mathbf{c}$ and $\mathbf{d}$ such that $\mathbf{b} \neq \mathbf{c}$. Suppose also that:

$$\mathrm{A}\mathbf{b} = \mathbf{d}$$
$$\mathrm{A}\mathbf{c} = \mathbf{d}$$

(a) Show that there must be some non-zero vector $\mathbf{e}$ such that:

$$\mathrm{A}\mathbf{e} = \mathbf{0}$$

(6 marks)

(b) Using the properties of linear independence, explain why the result of part B2(iii)(a) shows that the columns of A are *linearly dependent*.

(2 marks)

END OF SECTION B