

Name \_\_\_\_\_

Student Number \_\_\_\_\_

Instructor: S. Qiao

## CAS708/CES700 Midterm Examination

Duration of examination: One hour

**1. (5 marks)** Recall that in the IEEE single precision,  $\beta = 2$ ,  $t = 24$ ,  $e_{\min} = -126$ ,  $e_{\max} = 127$ , and the bias is 127. Give the IEEE single precision floating-point binary format

s eeeeeeee ffffffffffffffffffffffff

for each of the following floating-point numbers:

(a)  $-0.0$  Answer 1 00000000 0000000000000000000000(b)  $2^{-127}$  Answer 0 00000000 100000000000000000000000

(c) the unit of roundoff Answer 0 01100111 000000000000000000000000

(d) the largest number (not  $\infty$ ) Answer 0 11111110 111111111111111111111111

(e) NaN Answer 0 11111111 100000000000000000000000

**2. (8 marks)** Assume a floating-point number system characterized by  $\beta = 10$ ,  $t = 3$ ,  $e_{\min} = -10$ , and  $e_{\max} = 10$ ,

(a) give an example to show that the relative error in  $a \otimes b \oplus a \otimes c$  is at least 50%;

**Solution** Let  $a = 1.30$ ,  $b = 1.99$ , and  $c = -1.98$ , then  $a \otimes b \oplus a \otimes c = 2.59 - 2.57 = 2.00 \times 10^{-2}$  and  $a(b + c) = 1.3 \times 10^{-2}$ . The relative error is  $0.7/1.3 > 53\%$ .

(b) for the same  $a, b, c$ , what is  $a \otimes (b \oplus c)$ ?

**Solution** For the same  $a, b, c$ ,  $a \otimes (b \oplus c) = 1.30 \times 10^{-2}$ .

(c) the large relative error in  $a \otimes b \oplus a \otimes c$  is due to the unstable algorithm or the ill-conditioning problem? Justify your answer.

**Solution** The backward error analysis

$$\begin{aligned} & a \otimes b \oplus a \otimes c \\ &= (ab(1 + \epsilon_1) + ac(1 + \epsilon_2))(1 + \epsilon_3) \\ &= a(1 + \epsilon_3)(b(1 + \epsilon_1) + c(1 + \epsilon_2)), \end{aligned}$$

where  $|\epsilon_1|, |\epsilon_2|, |\epsilon_3| \leq u$ , shows that the algorithm is backward stable. Thus the large error must be due to the ill-conditioning problem.

**3. (3 mark)** Give an algorithm in Matlab style for computing the smallest positive floating-point number in the underlining system.

**Solution**

```
e = 1.0;
halfe = e/2;
while halfe > 0.0
    e = halfe;
    halfe = halfe/2;
end
e,
```

4. (4 marks) In IEEE standard, floating-point multiplication is a correctly rounded operation, that is,

$$a \otimes b = (ab)(1 + \epsilon), \quad |\epsilon| \leq u,$$

where  $u$  is the unit of roundoff. Is this true when one of  $a$  and  $b$  is a denormal? If your answer is yes, prove it. If your answer is no, give an example.

**Solution** No. Let  $a = 1.0\dots0_2 \times 2^{-1}$  and  $b = 0.0\dots0101_2 \times 2^{-126}$ , then  $a \otimes b = 0.0\dots010_2 \times 2^{-126}$  and the relative error  $\epsilon = (2^{-24} \times 2^{-126}) / (1.01 \times 2^{-22} \times 2^{-126}) > 2^{-3}$ .

5. (4 marks) In the Simpson's rule, the truncation error in a panel  $[x_i, x_{i+1}]$  is

$$I_i - S = \frac{1}{2880} h_i^5 f^{iv}(y_i) + \dots,$$

where  $h_i = x_{i+1} - x_i$  and  $y_i = (x_i + x_{i+1})/2$ . Derive a higher order quadrature by combining  $S$  and  $S_{1/2}$  (doubling the number of panels).

**Solution**

$$\begin{aligned} I_i - S_{1/2} &= \frac{1}{2880} \left(\frac{h_i}{2}\right)^5 \left(f^{iv}(x_i + h_i/4) + f^{iv}(x_i + 3h_i/4)\right) \\ &\approx \frac{1}{2880} \left(\frac{h_i}{2}\right)^5 (2f^{iv}(y_i)) \\ &= \frac{1}{16}(I_i - S). \end{aligned}$$

From  $I_i - S_{1/2} \approx \frac{1}{16}(I_i - S)$ , we have  $I_i \approx \frac{16}{15}S_{1/2} - \frac{1}{15}S$ .

**6. (5 marks)** Consider a two-dimensional function  $f(x, y)$  whose values are known at the distinct  $(x_i, y_j)$ ,  $i = 1, 2, 3$ ,  $j = 1, 2$ . Find the Lagrangian interpolation polynomial approximation  $P(x, y)$  of  $f(x, y)$ . Express  $P(x, y)$  in vector-matrix form.

(a) Fix  $y$ , let  $g(x) = f(x, y)$ , then interpolate  $g(x)$  at  $x_i$  and write the result in vector form.

**Solution**

$$\begin{aligned} g(x) &\approx \sum_{i=1}^3 f(x_i, y) \prod_{k \neq i} \frac{x - x_k}{x_i - x_k} \\ &= \begin{bmatrix} \frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)} & \frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)} & \frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)} \end{bmatrix} \begin{bmatrix} f(x_1, y) \\ f(x_2, y) \\ f(x_3, y) \end{bmatrix}. \end{aligned}$$

(b) For each  $f(x_i, y)$ , interpolate it at  $y_j$  and express it in vector form:

**Solution** Interpolating  $f(x_i, y)$  for each  $i$ , we have

$$\begin{aligned} f(x_i, y) &\approx \sum_{j=1}^2 f(x_i, y_j) \prod_{k \neq j} \frac{y - y_k}{y_j - y_k} \\ &= \begin{bmatrix} f(x_i, y_1) & f(x_i, y_2) \end{bmatrix} \begin{bmatrix} \frac{y-y_2}{y_1-y_2} \\ \frac{y-y_1}{y_2-y_1} \end{bmatrix}. \end{aligned}$$

(c) Putting all together in matrix-vector form:

**Solution**

$$P(x, y) = \begin{bmatrix} \frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)} & \frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)} & \frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)} \end{bmatrix} \begin{bmatrix} f(x_1, y_1) & f(x_1, y_2) \\ f(x_2, y_1) & f(x_2, y_2) \\ f(x_3, y_1) & f(x_3, y_2) \end{bmatrix} \begin{bmatrix} \frac{y-y_2}{y_1-y_2} \\ \frac{y-y_1}{y_2-y_1} \end{bmatrix}.$$