

CS2262 Spring 2007

Exam 1

Name:

1. A binary floating point representation of a number x can be written as

$$x = \sigma \times \bar{x} \times 2^e$$

Define the terms σ , \bar{x} and e , and explain the range of values they take and their relevance for the size and precision of a number when represented on a computer.

2. What is the *machine epsilon* for a floating point format? Estimate how large this is for IEEE single precision format.

3. Convert the following numbers from decimal to binary: (i) 10, (ii) 16.375 . Convert the following numbers from hexadecimal to binary: (i) DA9.301, (ii) 0.0F18 . You may find this table helpful for the second part.

0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
A	1010
B	1011
C	1100
D	1101
E	1110
F	1111

4. The definition of the Taylor polynomial of degree n for the function $f(x)$ about the point x_0 is

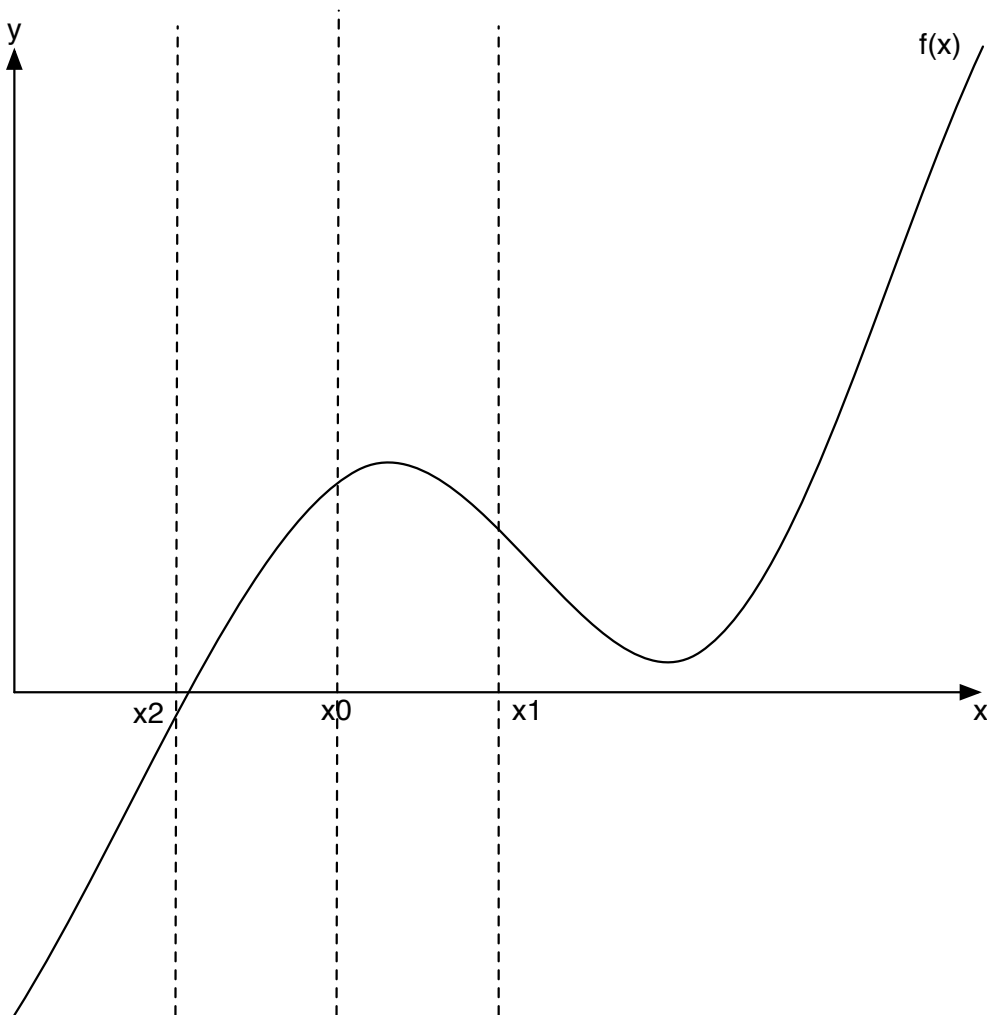
$$p_n(x) = \sum_{j=0}^n \frac{(x - x_0)^j}{j!} f^{(j)}(x_0)$$

The remainder term is defined by

$$R_n(x) = \frac{(x - x_0)^{n+1}}{(n + 1)!} f^{(n+1)}(c), \quad \alpha \leq x \leq \beta$$

where c lies somewhere between x_0 and x .

Draw a Taylor polynomial of degree 1 about the point $x = x_0$ for the function in the graph below. Illustrate on the graph the error in approximating the function by the Taylor polynomial at x_1 and x_2 . Use the form of the remainder term to explain the difference in the size of the error at x_1 and x_2 .



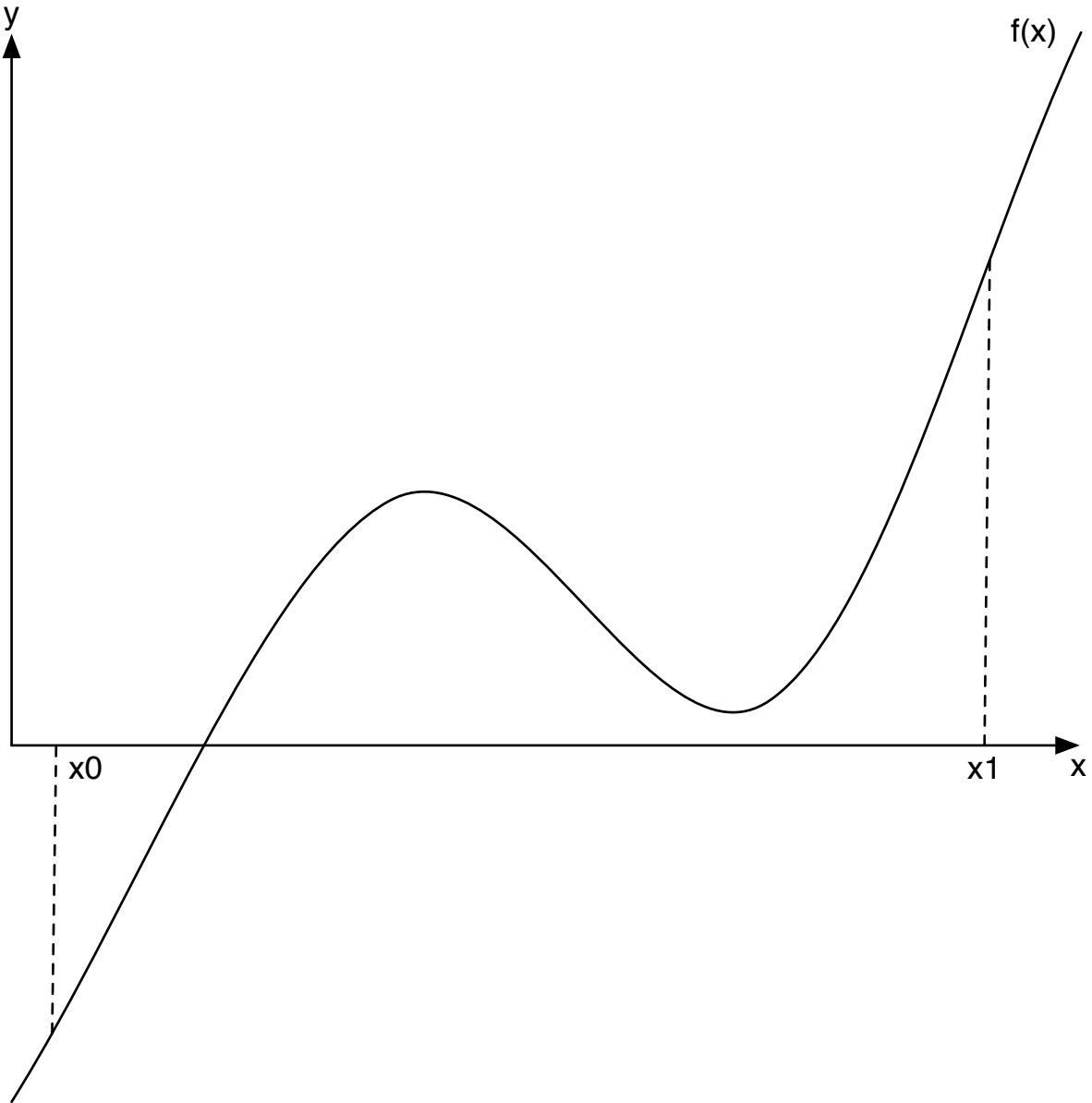
5. Calculate the 3rd degree Taylor polynomial about $x = 0$ for the functions (i) $\exp(x)$ (ii) $\log(1+x)$
(Take a look at the next question before answering this).

6. Calculate a general expression for the n th degree Taylor polynomial about $x = 0$ for the functions
(i) $\exp(x)$ (ii) $\log(1 + x)$

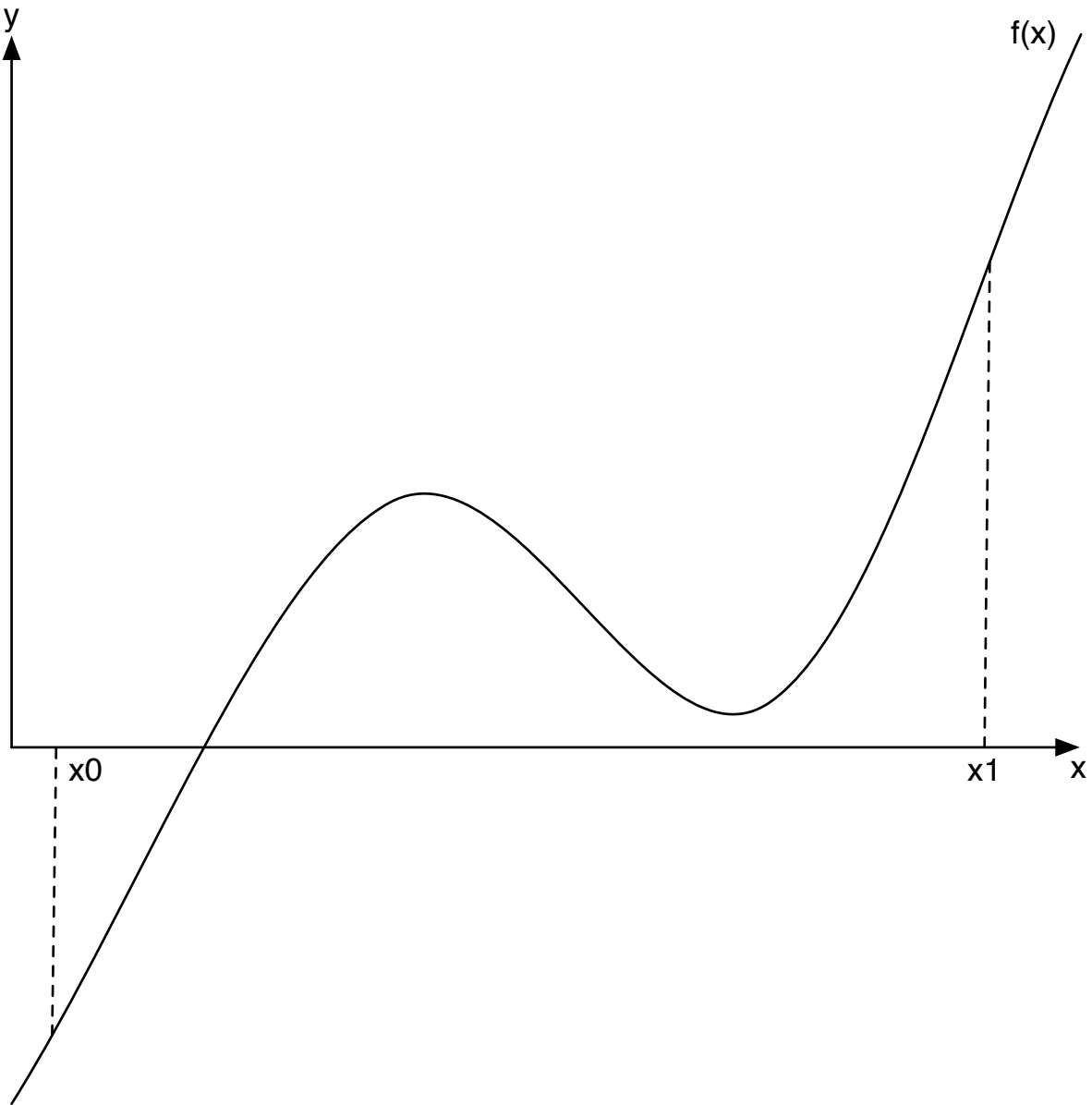
7. Calculate the remainder term for the functions in the last question, and estimate the degree of polynomial needed in each case for the size in the error at $x = 1$ to be less than 0.1. [The remainder term is defined in a previous question]

8. For what kind of values of x will $f(x) = \sqrt{x} - \sqrt{x-1}$ produce inaccurate results using floating point arithmetic (where $x > 1$). How can the function be reformulated to avoid this?

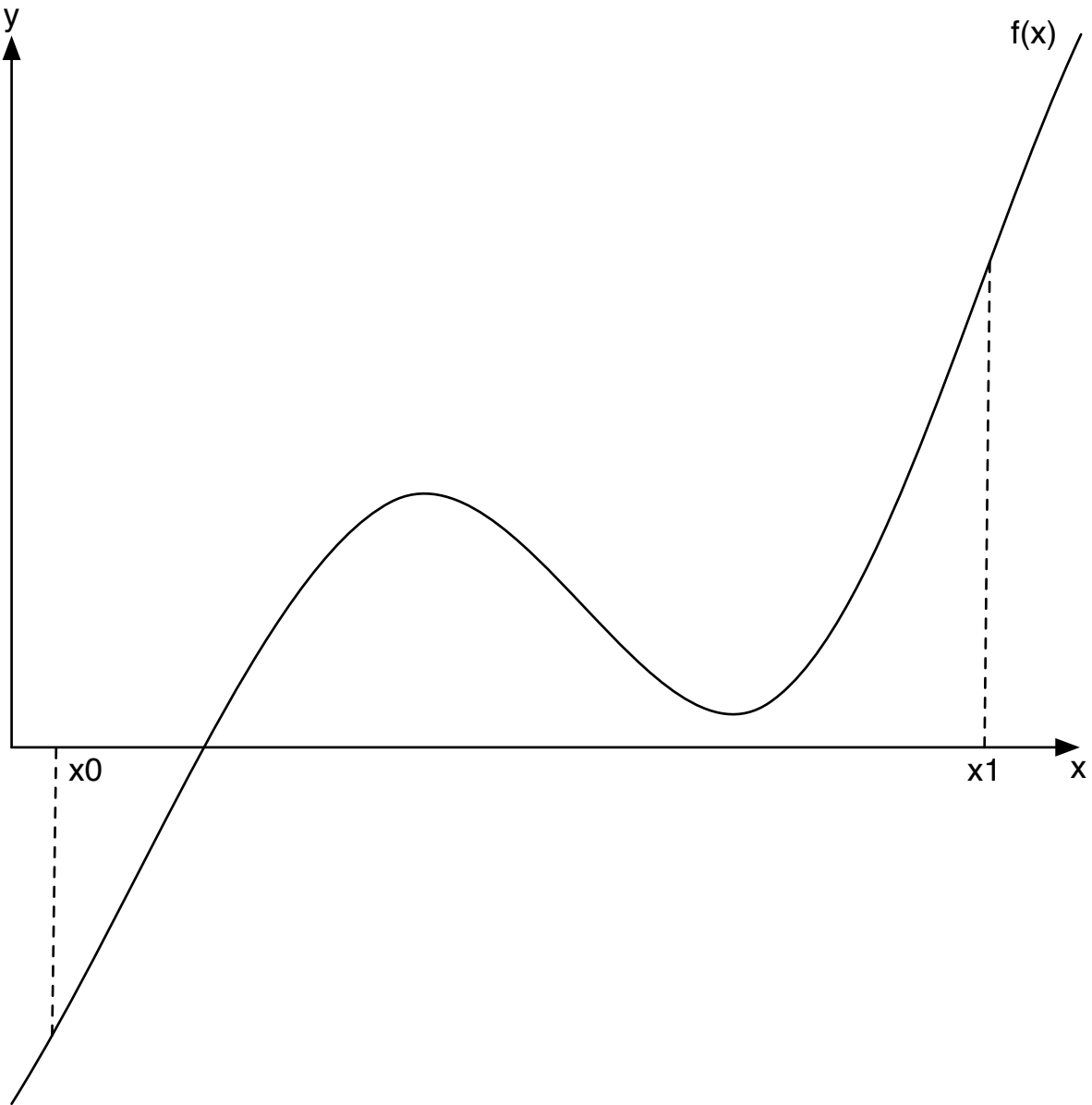
9. The following graph shows the function $f(x)$ with an initial bracket $[x_0, x_1]$. Using the bisection method algorithm starting from this initial bracket, draw on the graph successive estimates x_2, x_3, \dots for the root of $f(x) = 0$, stopping when you are close to the root, or the estimate diverges to a location outside of the graphed area.



10. The following graph shows the function $f(x)$ with an initial bracket $[x_0, x_1]$. Using the false position method algorithm starting from this initial bracket, draw on the graph successive estimates x_2, x_3, \dots for the root of $f(x) = 0$, stopping when you are close to the root, or the estimate diverges to a location outside of the graphed area.



11. The following graph shows the function $f(x)$ with an initial bracket $[x_0, x_1]$. Using the secant method algorithm starting from this initial bracket, draw on the graph successive estimates x_2, x_3, \dots for the root of $f(x) = 0$, stopping when you are close to the root, or the estimate diverges to a location outside of the graphed area.



12. The secant method has an error in computing the root α of a function $f(x)$ which converges with order 1.62. Explain what this means. Is this kind of convergence more or less desirable than linear convergence?

13. The following sequence of estimates is converging to a root at $p = 1.895494$. Estimate the order of convergence?

iteration	estimate
0	1.80000
1	1.85078
2	1.87375
3	1.88476
4	1.89016
5	1.89284
6	1.89417
7	1.89483
8	1.89516
9	1.89533
10	1.89541

14. The basic algorithm for the bisection method is

```
Input initial estimates a and b
BEGIN LOOP
  Calculate next estimate  $c = (a+b)/2$ 
  Estimate error, if below tolerance than stop
  If  $\text{func}(b) \times \text{func}(c) < 0$  then  $a = c$  else  $b = c$ 
END LOOP
```

Provide basic algorithms for both the false position and secant methods.