Solution Notes

(a) β is the base of the arithmetic.

p is the precision, i.e. the number of digits of base β in the significand, including the hidden bit.

e_{min}, e_{max} are the minimum, maximum values of the exponent e, where a normalised number is represented as

$$\pm \underset{\substack{\uparrow \\ \text{hidden bit}}}{1} . d_1 d_2 \ldots d_{p-1} \times \beta^e$$

There are 2046 values of e plus $e_{min}-1$, $e_{max}+1$ which are used for special purposes. This requires 11 bits for the exponent. The sign requires 1 bit and the significand $(p-1 =) 52$ bits, making (64 bits =) 8 bytes in total. [5 marks]

(b) The hidden bit is the most significant bit of the significand and can be deduced from the exponent, so need not be stored. It is 1 for normalised numbers, and 0 for denormal numbers. [2 marks]

(c) If x^* denotes the floating point representation of x, ϵ_m is the smallest positive number such that $(1+\epsilon_m)^* > 1$.

For IEEE Double Precision $\epsilon_m = 2^{-(p-1)} = 2^{-52}$. [3 marks]

(d) For computing $f'(x)$, we take $h = \sqrt{\epsilon_m} = 2^{-26}$, and expect an absolute error of $\sim h = 2^{-26}$.

For computing $f''(x)$, we take $h = \sqrt{2^{-26}} = 2^{-13}$, and expect an absolute error of $\sim h = 2^{-13}$. [4 marks]

(e) Assume $\beta = 2$. We require $f''(x)$ to be computed with an absolute accuracy of $10^{-3} \simeq 2^{-10}$. This implies that $f'(x)$ needs to be computed to an accuracy of 2^{-20} and that $f(x)$ needs to be stored with an accuracy of 2^{-40}.

Therefore we require $\epsilon_m = 2^{-(p-1)} = 2^{-40}$ so $p = 41$.

Taking account of the hidden bit, 40 bits are required for the significand, with 1 bit for the sign, leaving 7 bits for the exponent. The exponent can therefore take only 128 values, including $e_{min}-1$, $e_{max}+1$. So $e_{min} = -62$, $e_{max} = 63$. [6 marks]