Solutions 6

1. Two polynomial of degree at most $k-1$ can agree in at most $k-1$ points, so the Hamming distance between any two codes in the Reed-Solomon code is at least $n-k+1$. And clearly there are two polynomials of degree at most $k-1$ agreeing in exactly $k-1$ points, therefore the distance of the Reed-Solomon code is exactly $n-k+1$.

2. We can perform the standard division algorithm to divide the polynomial $q(Y) \in (\mathbb{F}[X])[Y]$ be the monic polynomial $Y-\beta$ with respect to the variable Y. We have

$$q(Y) = (Y-\beta)d(Y) + r(Y)$$

where $d(Y), r(Y) \in (\mathbb{F}[X])[Y]$ and the degree of $r(Y)$ is less than degree of $Y-\beta$ with respect to the variable Y. Thus $r(Y)$ must be a constant polynomial, i.e., $r(Y) \in \mathbb{F}[X]$. Plugging $Y=\beta$ in the equation above, we obtain $r = q(\beta)$ Therefore

$$q(\beta) = 0 \iff r = 0 \iff Y-\beta \text{ divides } q(Y).$$

3. Let $0 \leq p < \frac{1}{2}$ be the new constant replacing $\frac{1}{3}$ in the definition of the class **BPP**. Let P be the probabilistic algorithm that is wrong with probability p, and let m be a sufficiently large constant (we compute the value of m as a function of p later). Let P' be the algorithm that outputs the majority of the answers among m independent runs of the algorithm P. Let us estimate the probability that P' is wrong. The probability of P' being wrong is

$$\mathbb{P}[\text{Bin}(m,p) \geq \frac{m}{2}] \leq e^{-mp(\frac{1}{2p}-1)^2/3}.$$

Let $m = \frac{3\log(3)}{p\left(\frac{1}{2p}-1\right)^2}$. Then clearly P' is wrong with probability at most $\frac{1}{3}$. Moreover, if P needs r random bits, then P' needs $mr = O(r)$ random bits. Furthermore if P runs in polynomial time, then P' runs in polynomial time as well.