a) Given a non-negative integer *x* and a positive integer *y*, we say that *q* is the quotient and *r* the remainder when x is divided by *y* if *q*, the quotient is the result obtained and *r*, the remainder is the number that is leftover. Such that when x divided by y, x = qy + r for a quotient q and remainder r with r < y. And then the correctness of the recurrence could be proved by a case-analysis.

For the case x = 0, the result<0,0> is indeed correct as 0 divided by anything would result in a quotient and remainder of 0.

For the case x > 0, x is even and 2r′ < y, the recurrence shows the result as <2q′,2r′>. When x divided by y, x = qy + r for some quotient q and remainder r. Since x is even, $\lfloor x/2 \rfloor = x/2$. So x/2 = q′y + r′ → x = 2q′y + 2r′ for some quotient q′ and remainder r′. With these 2 equations x = qy + r and x = 2q′y + 2r′, it's obvious that 2q′ = q and 2r′ = r when 2r′ < y. Hence the quotient and remainder for x divided by y is indeed <2q′,2r′> when x > 0 and even and 2r′ < y.

For the case x > 0, x is even and 2r′ ≥ y, the recurrence shows the result as <2q′+1, 2r′–y>. When x divided by y, x = qy + r for some quotient q and remainder r. Since x is even, $\lfloor x/2 \rfloor = x/2$. So x/2 = q′y + r′ → x = 2q′y + 2r′ for some quotient q′ and remainder r′. With these 2 equations x = qy + r and x = 2q′y + 2r′, last case talks about the situation when 2r′ < y, so for this case, when 2r′ ≥ y, it has to subtract y in order to get a value smaller than y, as that's required property of a remainder, and in this case, it means that when x divided by y, it could subtract y for (2q′ + 1) times before having a value smaller than 0. Hence for this case, the quotient and remainder for x divided by y is <2q′+1, 2r′–y> when x > 0 and even and 2r′ ≥ y, which is indeed the same as the recurrence shows.

For the case x > 0, x is odd and 2r′ + 1 < y, the recurrence shows the result as <2q′,2r′+1>. When x divided by y, x = qy + r for some quotient q and remainder r. Since x is odd, $\lfloor x/2 \rfloor = (x-1)/2$. So (x – 1)/2 = q′y + r′ → x – 1 = 2q′y + 2r′ → x = 2q′y+2r′+1for some quotient q′ and remainder r′. Since the plus 1 is a constant, it needs to be seen as a remainder with 2r′ as a whole. So, for this case, it's obvious that 2q′ = q and 2r′ + 1 = r when 2r′ + 1 < y. Hence the quotient and remainder for x divided by y is indeed <2q′,2r′+1> when x > 0 and odd and 2r′+1 < y.

For the case that none of the above cases applies, the recurrence shows the result as <2q′+1,2r′+1-y>. When x divided by y, x = qy + r for some quotient q and remainder r. Since x is non-negative integer and y is a positive integer, the only possible situation is when x > 0 and x is odd and 2r′ + 1 ≥ y. Since x is odd, $\lfloor x/2 \rfloor = (x-1)/2$.
So (x – 1)/2 = q′y + r′ → x – 1 = 2q′y + 2r′ → x = 2q′y + 2r′ + 1. If 2r′ + 1 ≥ y, it needs to subtract y in order to get a value smaller than y, as that's required property of a remainder, and in this case, it means that when x divided by y, it could subtract y for (2q′ + 1) times before having a value smaller than 0. Hence for this case, the quotient and remainder for x divided by y is <2q′+1,2r′+1–y>, which is the same as the recurrence shows.

b) For this recursive algorithm, it uses the divide-n-conquer strategy, and this algorithm has time efficiency $T(n) = T(\lfloor n/2 \rfloor) + \Theta(\mathrm{n})$. And for convenience, we'll omit the floors, ceilings and boundary conditions. Hence, $T(n) = T(n/2) + \Theta(\mathrm{n})$.

By using the master method, for this recurrence, we have a = 1, b = 2, f(n) = n, and $n^{\log_b a} = n^{\log_2 1} = n^0 = 1$. Since $f(n) = \Omega(n^{\log_2 1+\epsilon})$, where ε = 1, we can apply case 3 of the master theorem if we can show that the regularity condition holds for f(n). For sufficiently large n, $af(n/b) = n/2 = cf(n)$ for c = 1/2. Consequently, by case 3, the solution to the recurrence is $T(n) = \Theta(\mathrm{n})$.