1a)
In order to obtain the smallest possible double value > 1 we can simply set the sign bit to 0 to obtain a positive value, then set the exponent bits to the decimal value of 0 since we can't obtain any value between close to above 1 whilst the exponent term is not equal to 1. And lastly we need to set the significand to it's lowest possible value this results in the following string of bits and this decimal value:

$$
\begin{aligned}
0 \quad & 01111111111 \quad 0000000000000000000000000000000000000000000000000001_b \\
& \mathrel{\widehat{=}} \quad (1 + 2^{-52}) \cdot 2^0 \approx 1.0000000000000002
\end{aligned}
$$

1b)
The smallest double value that you can add to 1 and not equal 1 is the following:

$$
\begin{aligned}
0.00000000000000012 + 1 &= 1.0000000000000002 \\
0.00000000000000011 + 1 &= 1
\end{aligned}
$$

Sadly I do not have a proper explanation for this since I simply used an algorithm to figure this out, but it probably has something to do with the rounding.

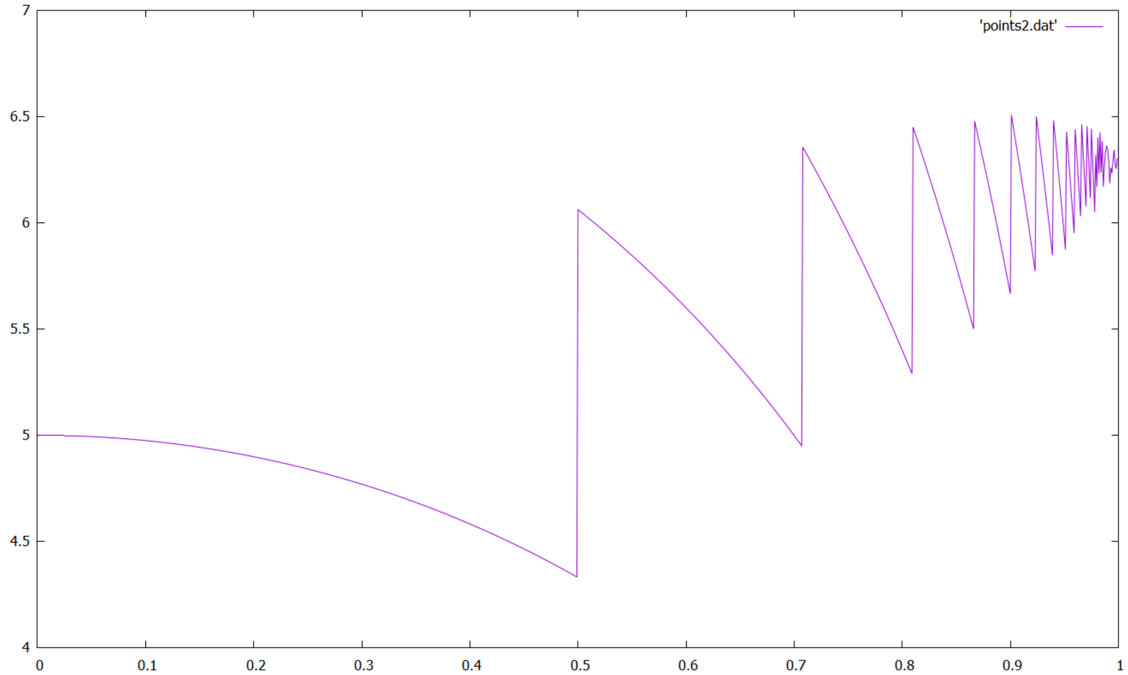
1c)
Similarly to 1a) we can simply choose the lowest possible values again, this time we can also use the lowest value for the exponent bits:

$$
\begin{aligned}
0 \quad & 00000000000 \quad 0000000000000000000000000000000000000000000000000001_b \\
& \mathrel{\widehat{=}} \quad 2^{-52} \cdot 2^{-1022} \approx 4.9406564584124654 \cdot 10^{-324}
\end{aligned}
$$

2a)
—

2b)
The higher y_0 gets the shallower the angle of impact is. This causes the amount of impacts along the edge of the circle to increase and with high enough y_0 even appoximate the circumference of the circle itself which would be equal to 2π. This is very similar to the analogy of trying to use same sided polygons to appoximate the circumference of a circle, since when looking at the path the point takes it looks very similar to same sided polygons increasing in amount of edges with growing y_0. Thus for a $y_0 \to 1$ you'd expect $|C| \to 2\pi$ like a polygon with infinite sides.

2c)
The program I've written, in addition to giving out the values of $|C|$ for $y_0 = 0.9$, $y_0 = 0.999$, also outputs a dataset which describes the length of C as a function of y_0. This can be plotted and results in the following graph:



As expected in 2b) the length of $|C|$ approximates 2π when $y_0 \to 1$. Additionally these are the expected console outputs of the program:

Curve length for y_0 = 0.9: 5.666569
Curve length for y_0 = 0.999: 6.304135