For this problem, note that if you worship Bessie, worshipping all T seconds in one session is always at least as good as worshipping Bessie in multiple sessions, since splitting up sessions would divide the segments' times and greatly reduce the moo points gained.

Easy Version:

Since times are between -3*10^4 and 3*10^4, this is a small enough range that we can iterate over it directly.

Sort the people by the time they join the line. At each point in time, update how many seconds are required to reach the front of the line if you were to join the line right now, calculate the moo points you would earn if you joined the line at this moment in time (current_time seconds of training before joining the line + S-current_time-line_time-T seconds of training after worshipping = current_time^2 + (S-current_time-line_time-T)^2 moo points.), and update maximum.

This solution's complexity is O(max_time + N log N).

Hard Version:

Times have magnitude <= 10^9, so we can no longer iterate over the times.

The solution idea is the same as the easy version, but we only consider the important points in time, which are the moments when a person is added, when a line becomes empty, t = 0, and t = S-T.

No other points are important because:

1. If we're adding ourselves to the line before another person is added & the line isn't empty, it is more optimal to just train for that duration and wait until the last second to join the line.
2. If we're adding ourselves to the line when the line is empty, it's always best to do it either right at the beginning of this empty line interval or right at the end (which is covered under the moment when a person is added), because the value of a^2 + (S-T-a)^2 is maximized when a is either really small or really large, so it's never optimal to pick an a that is between the minimum possible a and the maximum possible a.

This solution's complexity is O(N log N).