This is the DAG scheduling problem, which is NPC [2, 1].
We can use bitmask DP. Sort the courses in inverse topological order. Let $f[i][j]$ denote the minimum (x, y) such that we can use x days and select y courses on the last day to take courses set i, and take course j on the last day (its prerequisites cannot be taken on the same day). $O(2^n \cdot n)$.

Remark. the test cases are weak, and many of the "polynomial time" algorithms in the discussion forum are incorrect.

References

[1] Yu-Kwong Kwok and Ishfaq Ahmad. Static scheduling algorithms for allocating directed task graphs to multiprocessors. *ACM Computing Surveys (CSUR)*, 31(4):406–471, 1999.

[2] Jeffrey D. Ullman. Np-complete scheduling problems. *Journal of Computer and System sciences*, 10(3):384–393, 1975.