

Earliest-Start-Time-First Demo

CS 3AC3

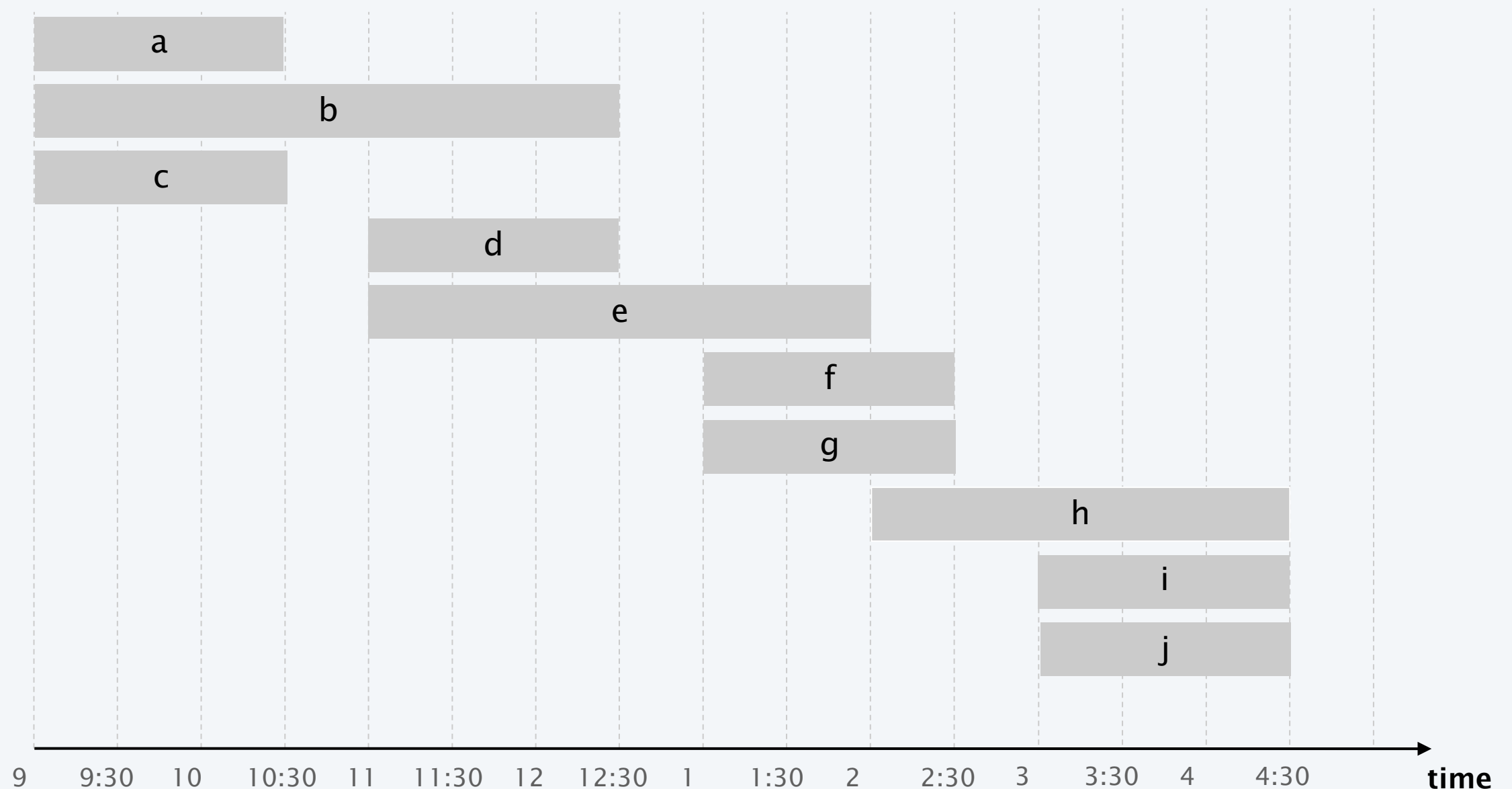
Ryszard Janicki

Department of Computing and Software, McMaster University, Hamilton,
Ontario, Canada

Earliest-start-time-first algorithm demo

Consider lectures in order of start time:

- Assign next lecture to any available classroom (if one exists).
- Otherwise, open up a new classroom.

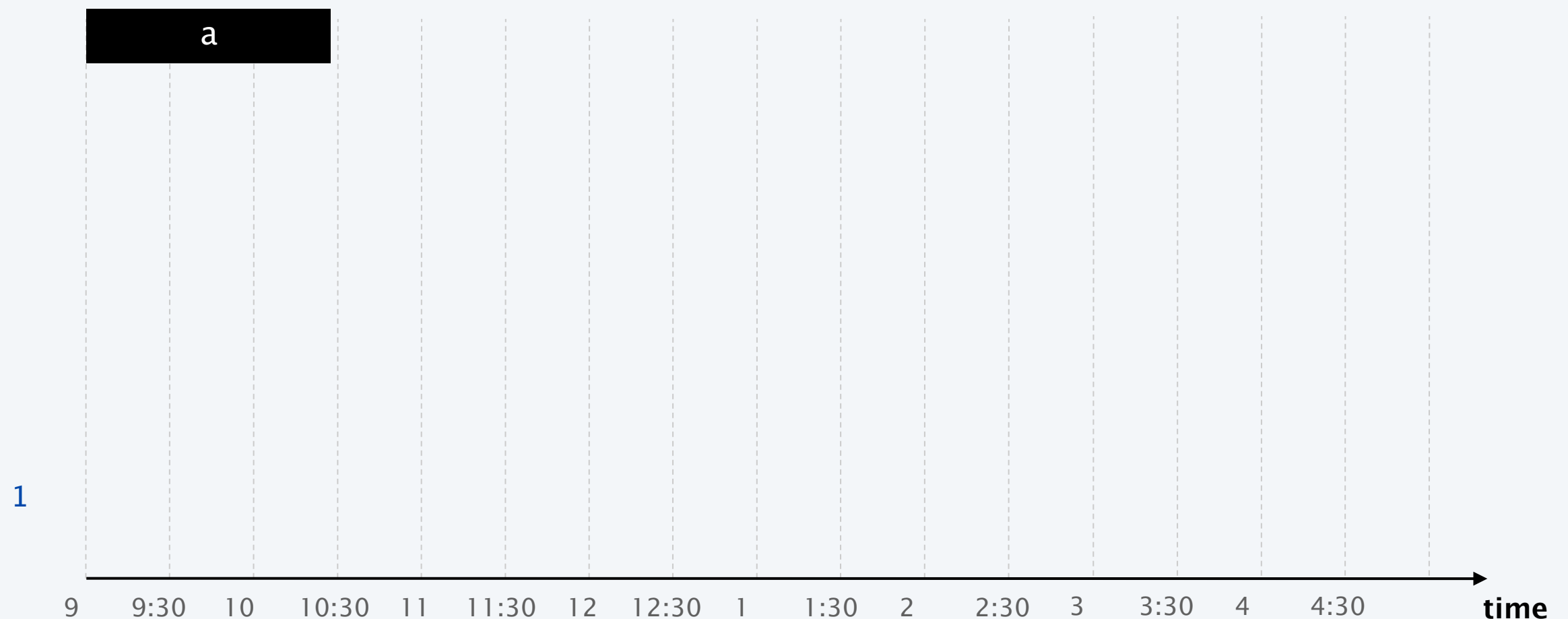


Earliest-start-time-first algorithm demo

Consider lectures in order of start time:

- Assign next lecture to any available classroom (if one exists).
- Otherwise, open up a new classroom.

no available classroom: open up a new classroom and assign lecture to it

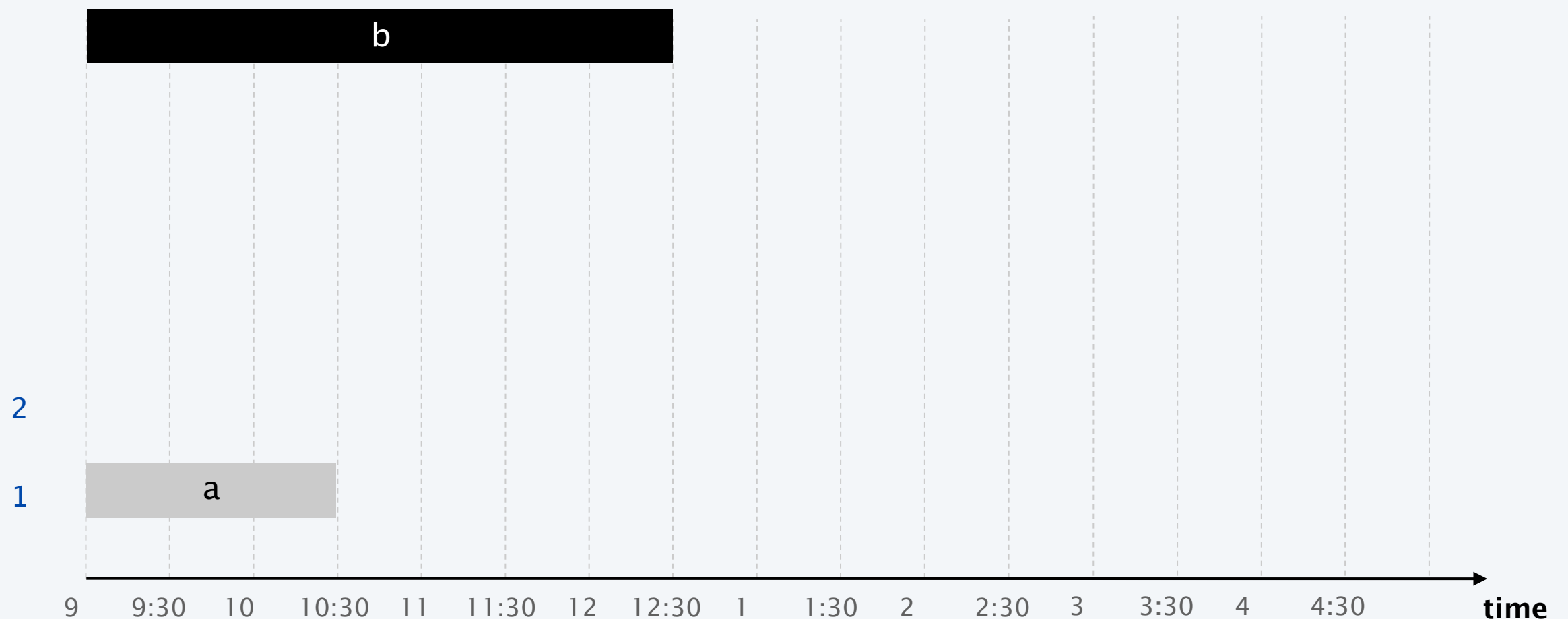


Earliest-start-time-first algorithm demo

Consider lectures in order of start time:

- Assign next lecture to any available classroom (if one exists).
- Otherwise, open up a new classroom.

no available classroom: open up a new classroom and assign lecture to it

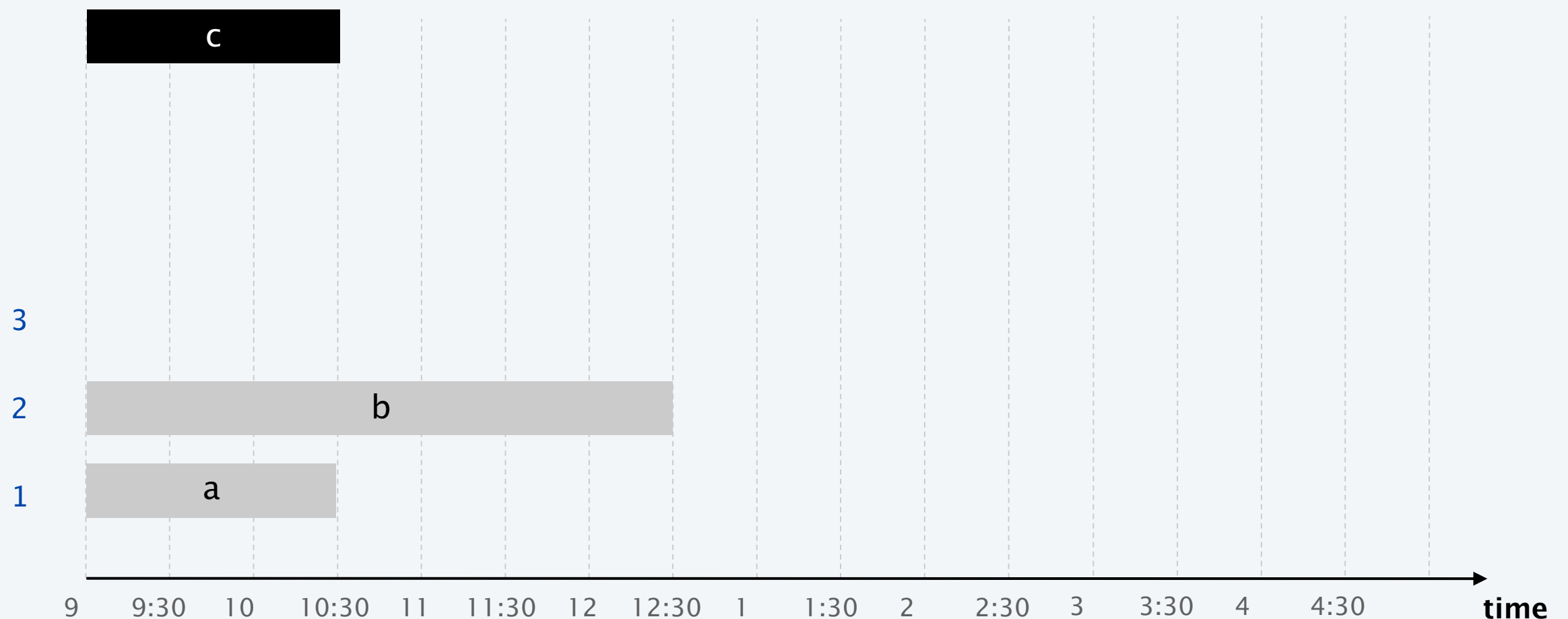


Earliest-start-time-first algorithm demo

Consider lectures in order of start time:

- Assign next lecture to any available classroom (if one exists).
- Otherwise, open up a new classroom.

no available classroom: open up a new classroom and assign lecture to it

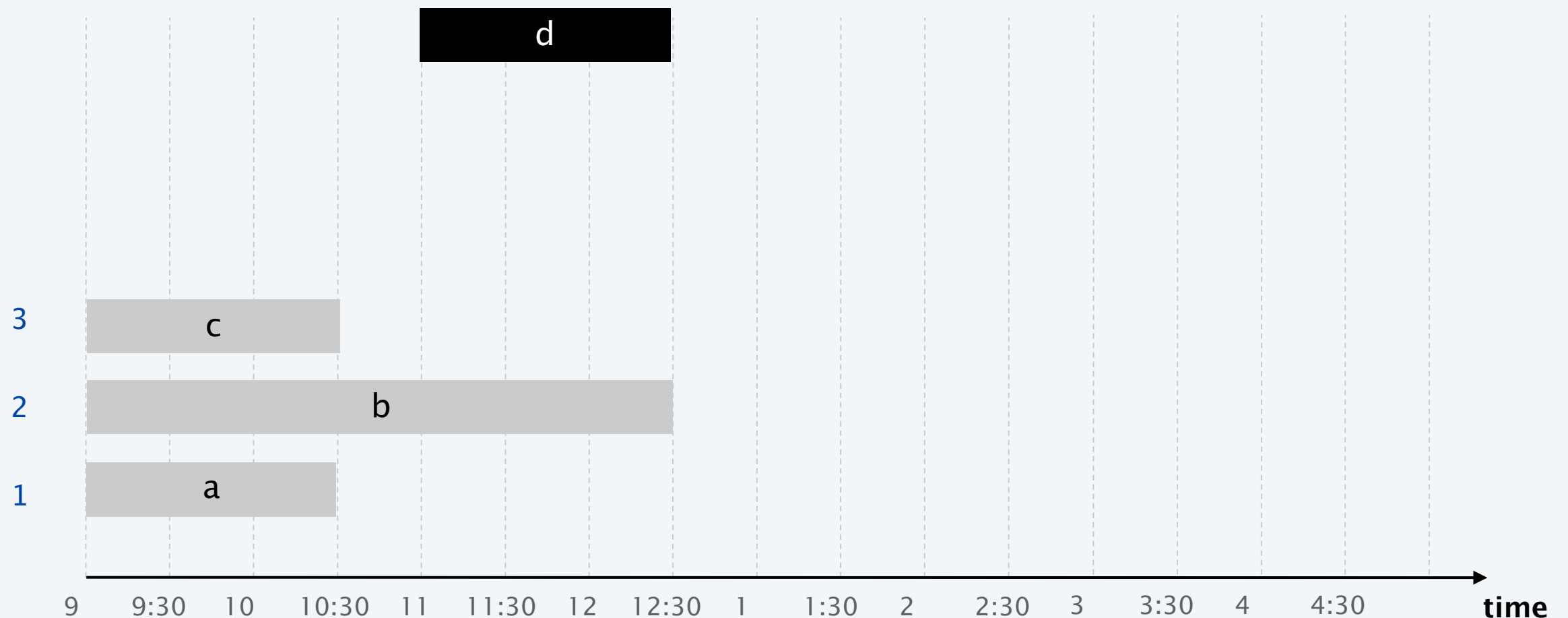


Earliest-start-time-first algorithm demo

Consider lectures in order of start time:

- Assign next lecture to any available classroom (if one exists).
- Otherwise, open up a new classroom.

lecture d is compatible with classrooms 1 and 3

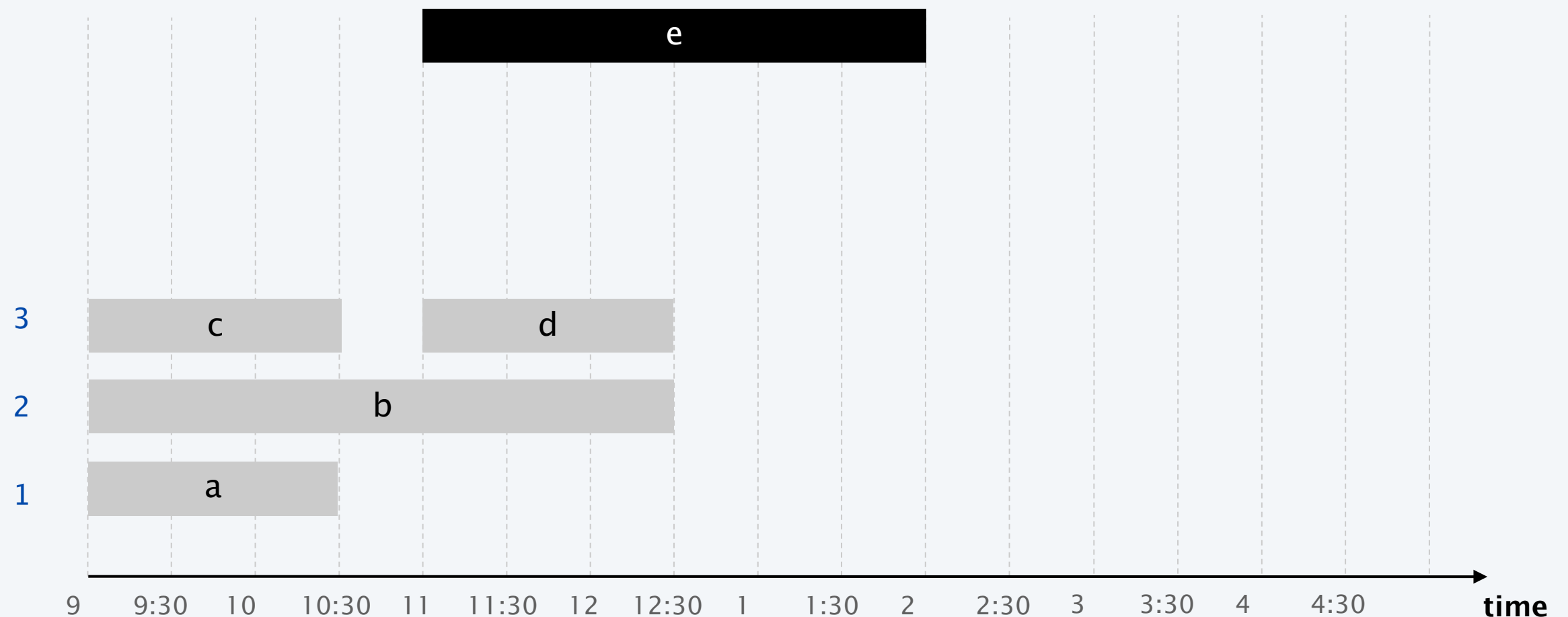


Earliest-start-time-first algorithm demo

Consider lectures in order of start time:

- Assign next lecture to any available classroom (if one exists).
- Otherwise, open up a new classroom.

lecture e is compatible with classroom 1

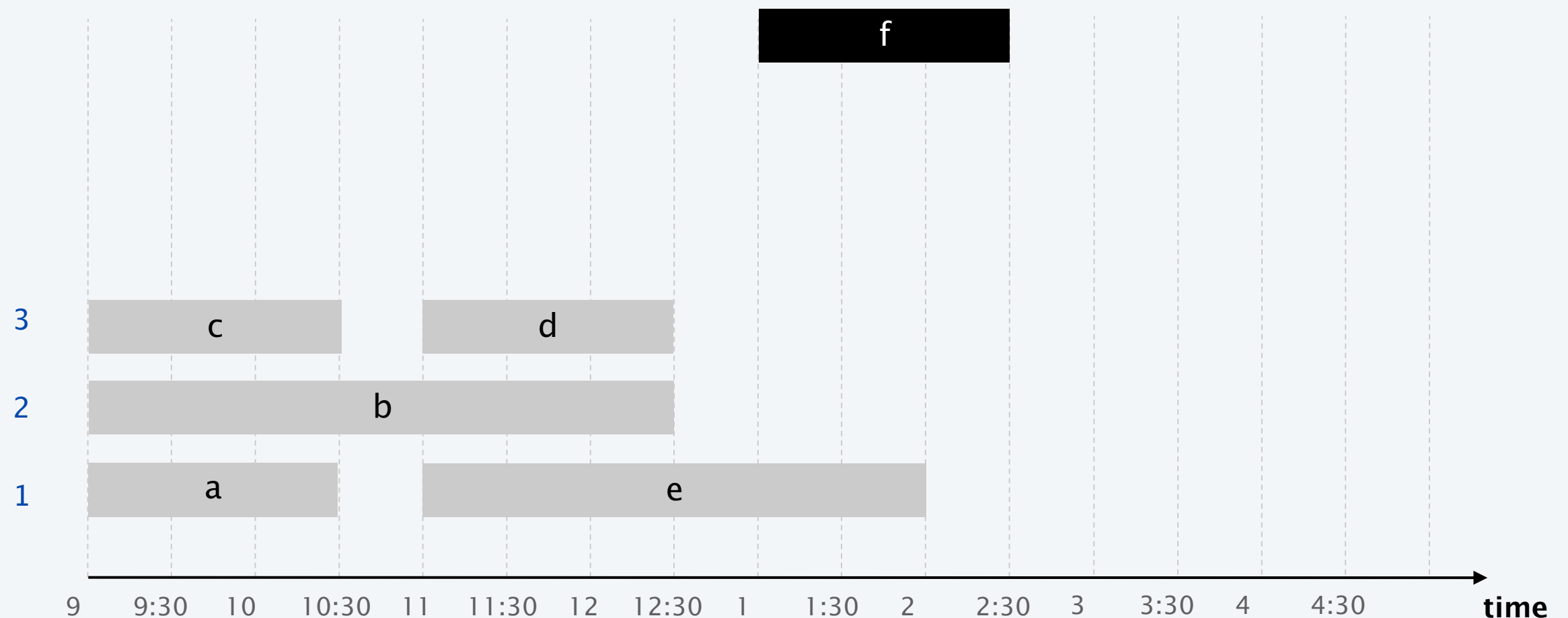


Earliest-start-time-first algorithm demo

Consider lectures in order of start time:

- Assign next lecture to any available classroom (if one exists).
- Otherwise, open up a new classroom.

lecture f is compatible with classroom 2 and 3

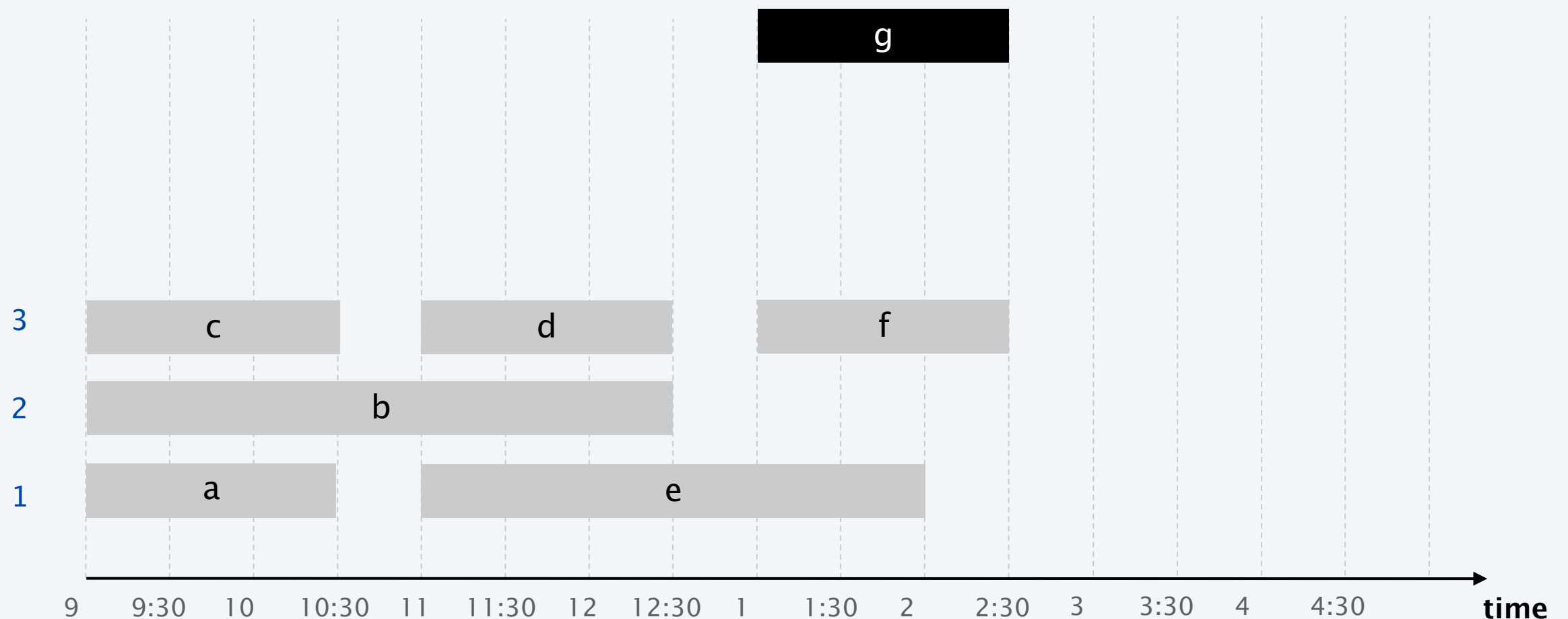


Earliest-start-time-first algorithm demo

Consider lectures in order of start time:

- Assign next lecture to any available classroom (if one exists).
- Otherwise, open up a new classroom.

lecture g is compatible with classroom 2

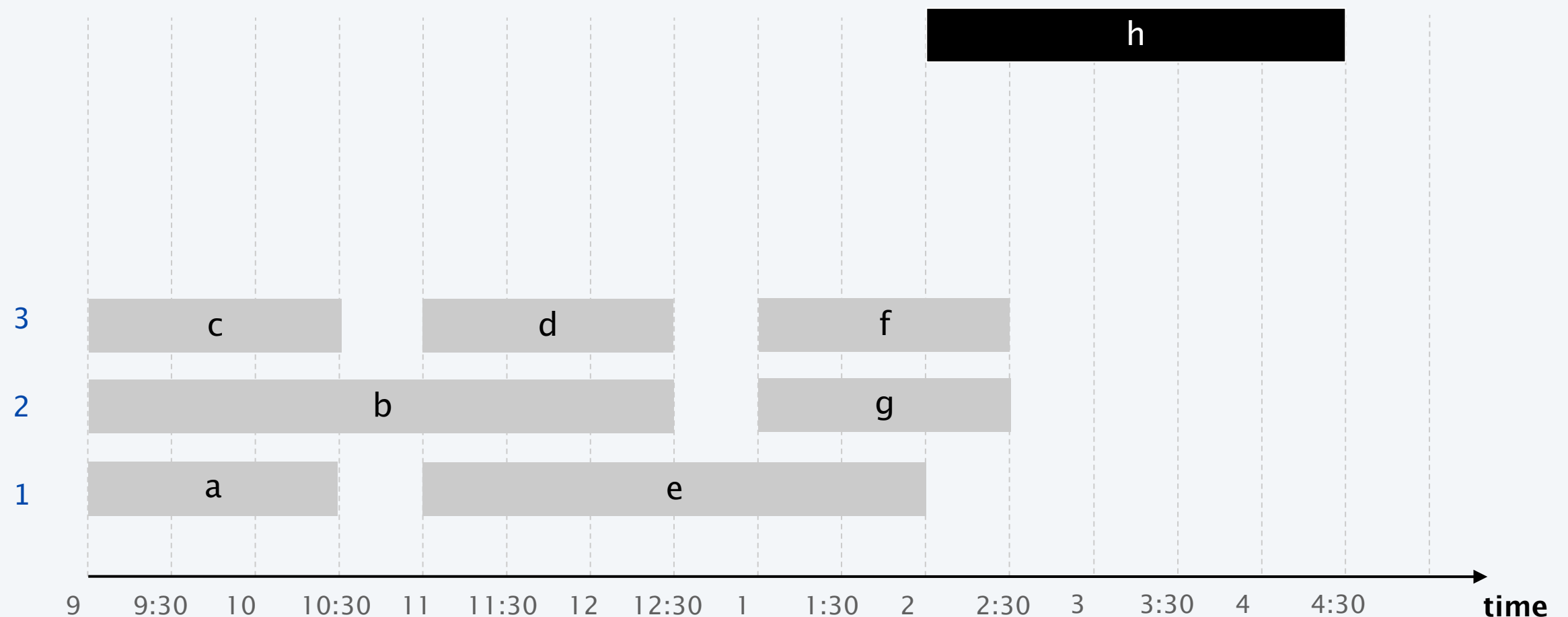


Earliest-start-time-first algorithm demo

Consider lectures in order of start time:

- Assign next lecture to any available classroom (if one exists).
- Otherwise, open up a new classroom.

lecture h is compatible with classroom 3

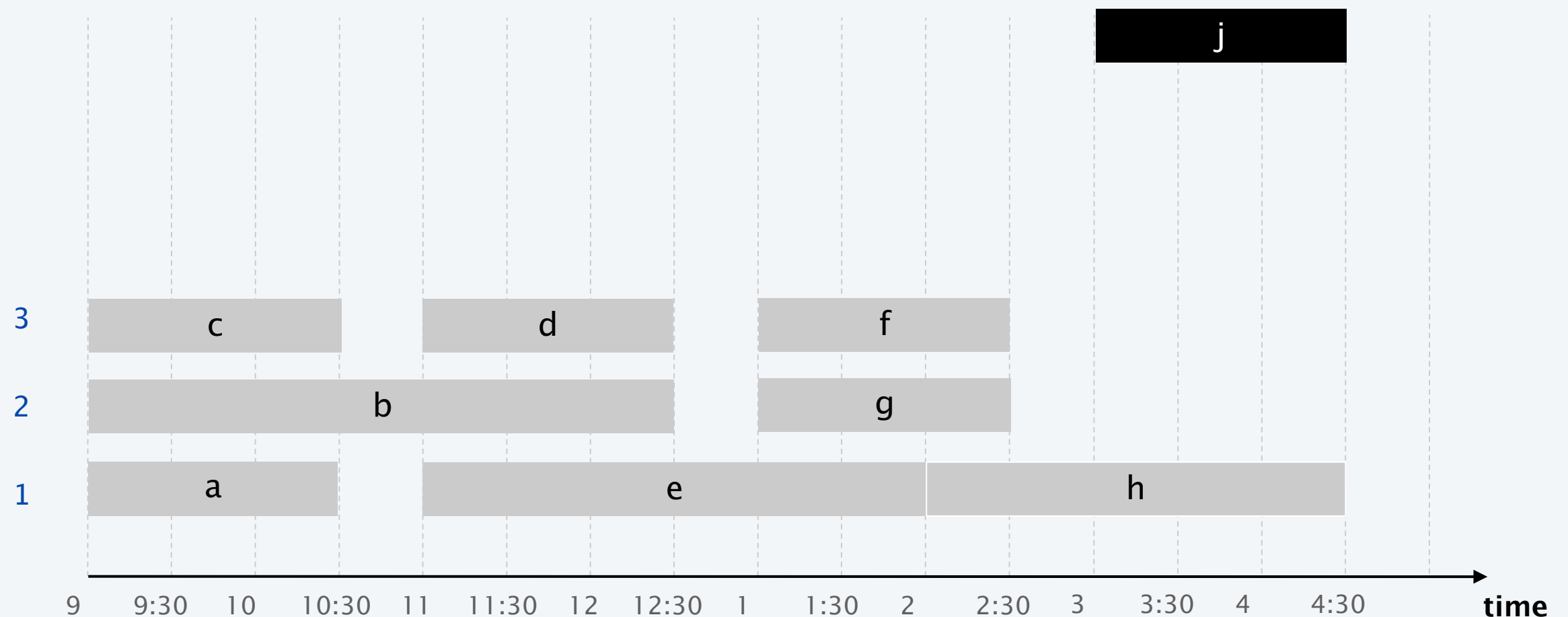


Earliest-start-time-first algorithm demo

Consider lectures in order of start time:

- Assign next lecture to any available classroom (if one exists).
- Otherwise, open up a new classroom.

lecture j is compatible with classrooms 2 and 3

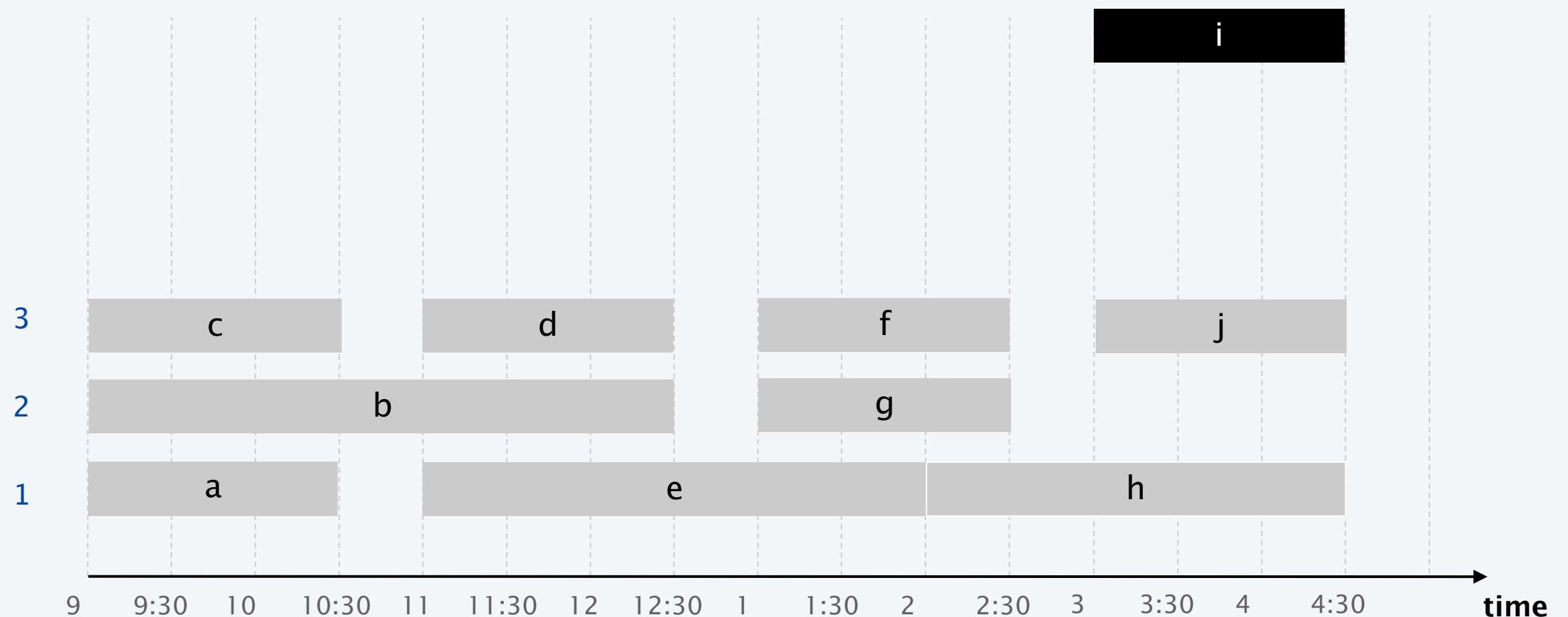


Earliest-start-time-first algorithm demo

Consider lectures in order of start time:

- Assign next lecture to any available classroom (if one exists).
- Otherwise, open up a new classroom.

lecture i is compatible with classroom 2



Earliest-start-time-first algorithm demo

Consider lectures in order of start time:

- Assign next lecture to any available classroom (if one exists).
- Otherwise, open up a new classroom.

done

