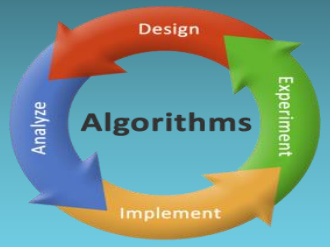
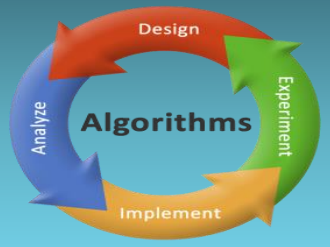
Greedy algorithms

Activity selection



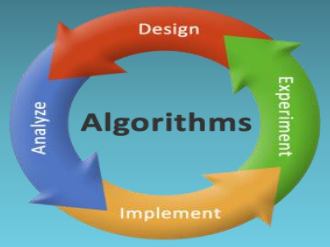
An Activity Selection Problem (Conference Scheduling Problem)

- **Input: A set of activities S = {a_1,…, a_n}**
- Each activity has start time and a finish time
  - ✓ a_i=[s_i, f_i)
- Two activities are compatible if and only if their interval does not overlap
- **Output: a maximum-size subset of mutually compatible activities**



Optimum structure

- Let us denote by S_{ij} the set of activities that start after activity a_i finishes and that finish before activity a_j starts
- and the maximum set of mutually compatible activities in S_{ij} by A_{ij}
- $$c[i,j] = \begin{cases} 0 & S_{ij} = \emptyset \\ \max_k \{c[i,k] + c[k,j] + 1\} & S_{ij} \neq \emptyset \end{cases}$$
- $c[0, n+1]$?

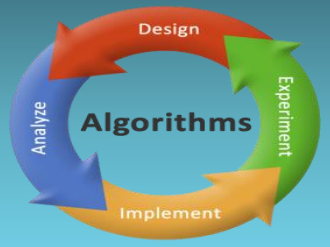


The Activity Selection Problem

- Here are a set of start and finish times

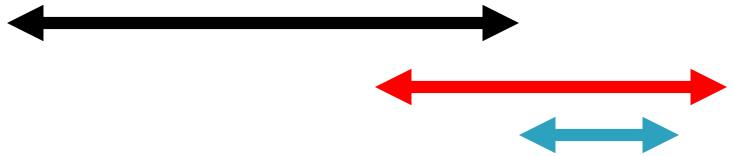
i	1	2	3	4	5	6	7	8	9	10	11
s_i	1	3	0	5	3	5	6	8	8	2	12
f_i	4	5	6	7	8	9	10	11	12	13	14

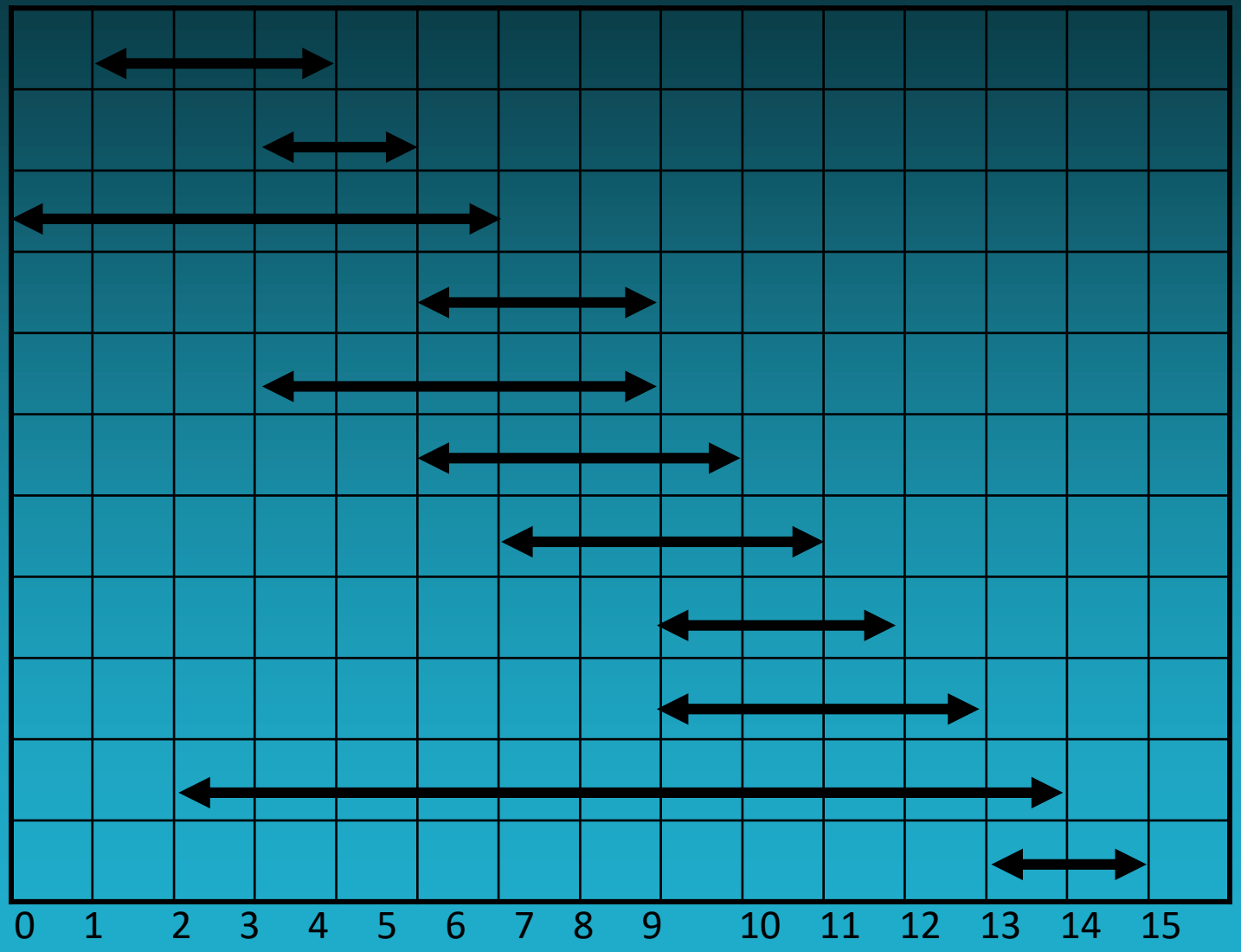
- What is the maximum number of activities that can be completed?
  - ✓ $\{a_3, a_9, a_{11}\}$ can be completed
  - ✓ But so can {a1, a4, a8' a11} which is a larger set
  - ✓ But it is not unique, consider {a2, a4, a9' a11}

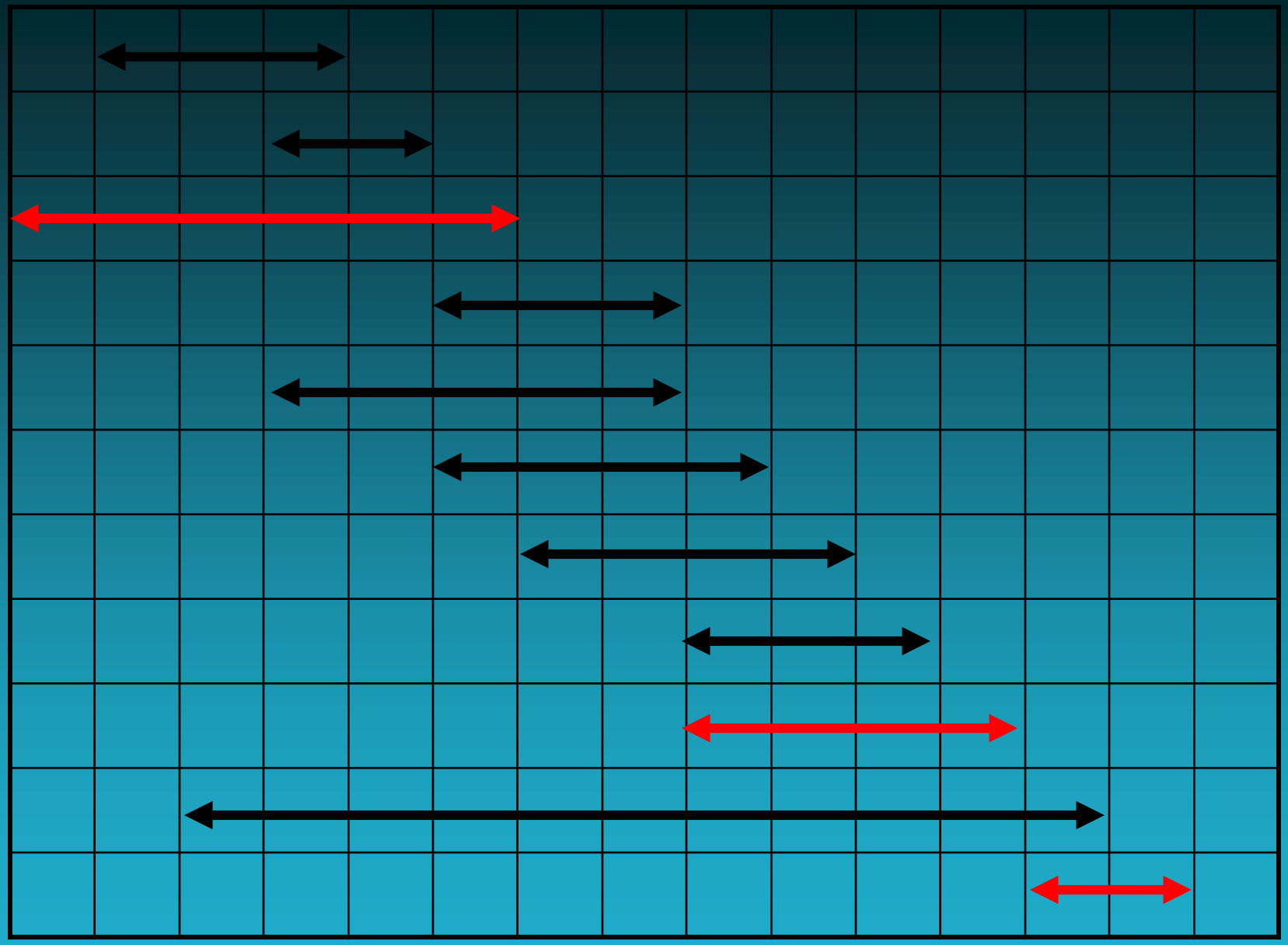


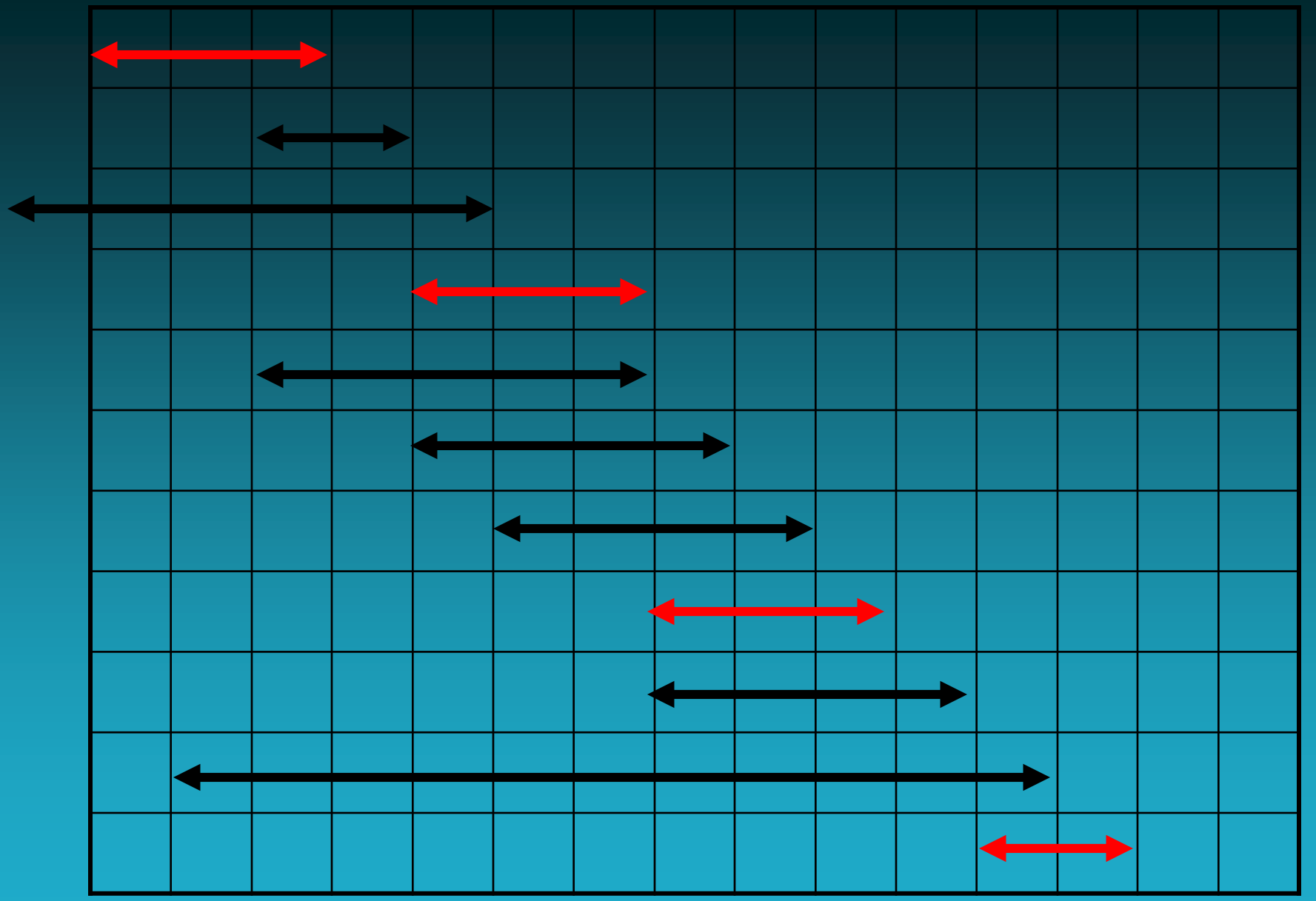
Interval Representation

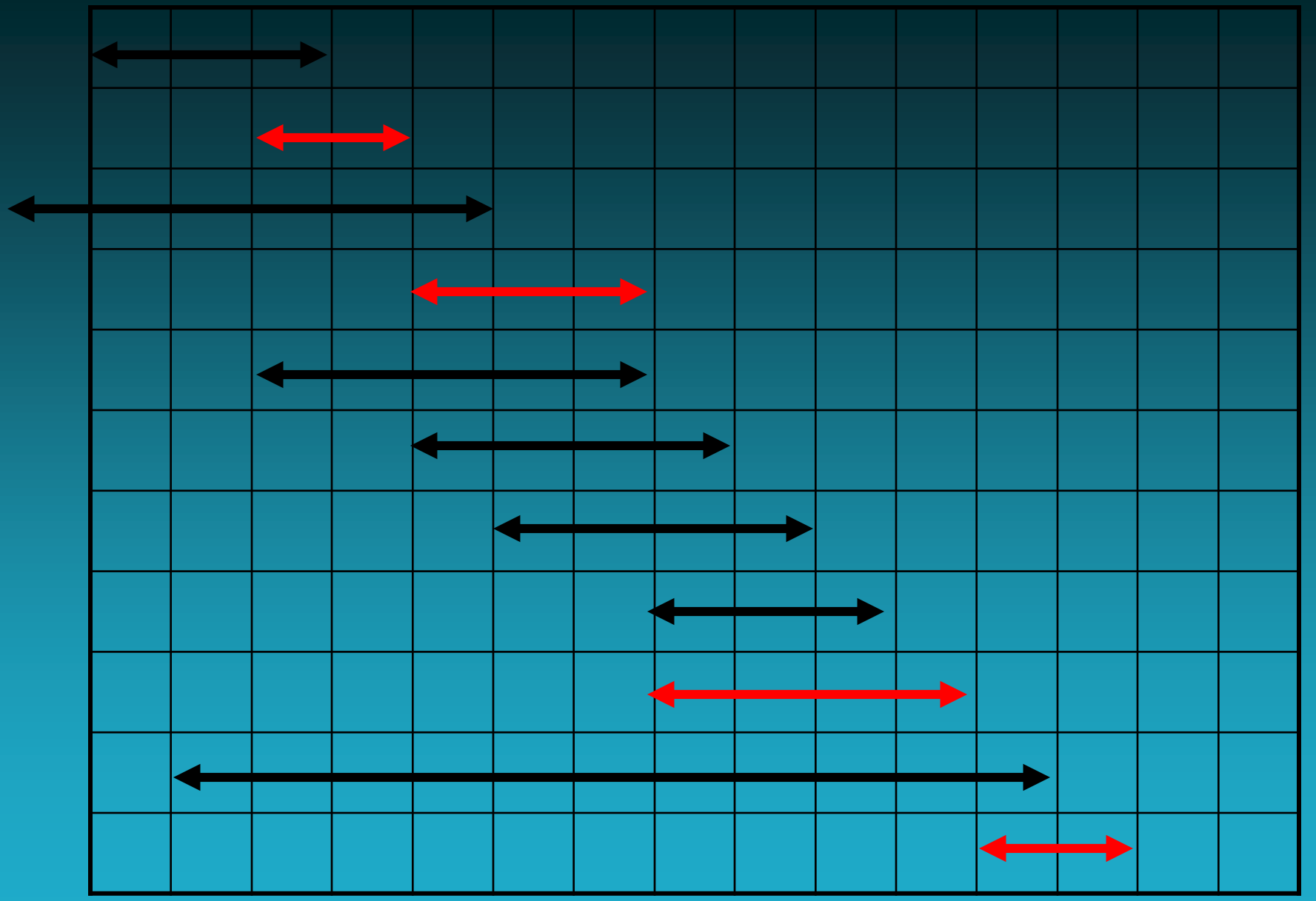
i	1	2	3	4	5	6	7	8	9	10	11
s_i	1	3	0	5	3	5	6	8	8	2	12
f_i	4	5	6	7	8	9	10	11	12	13	14

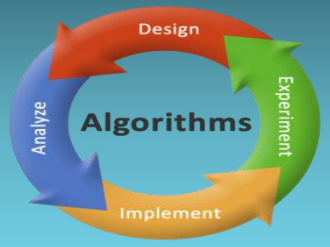



0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15



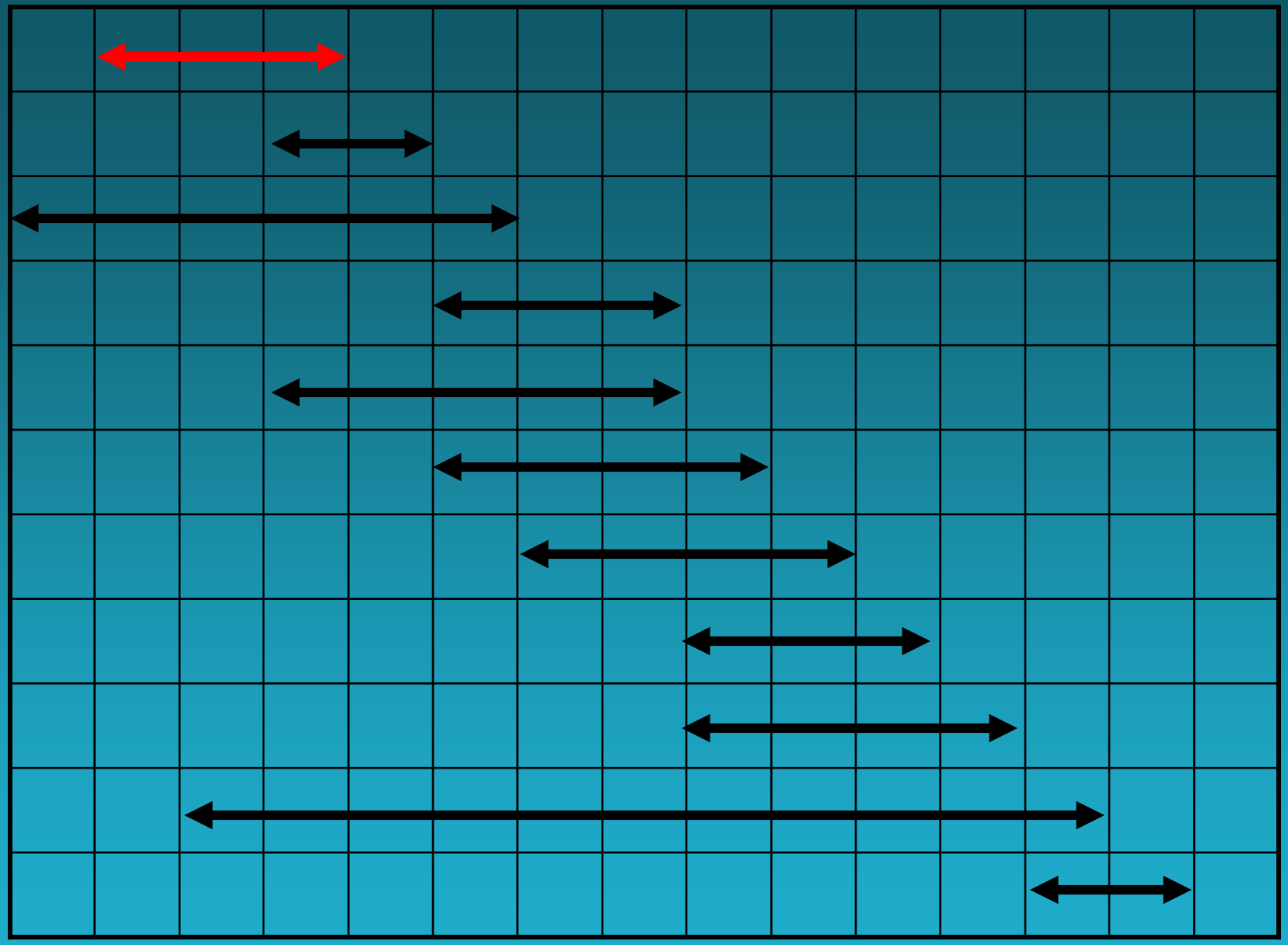


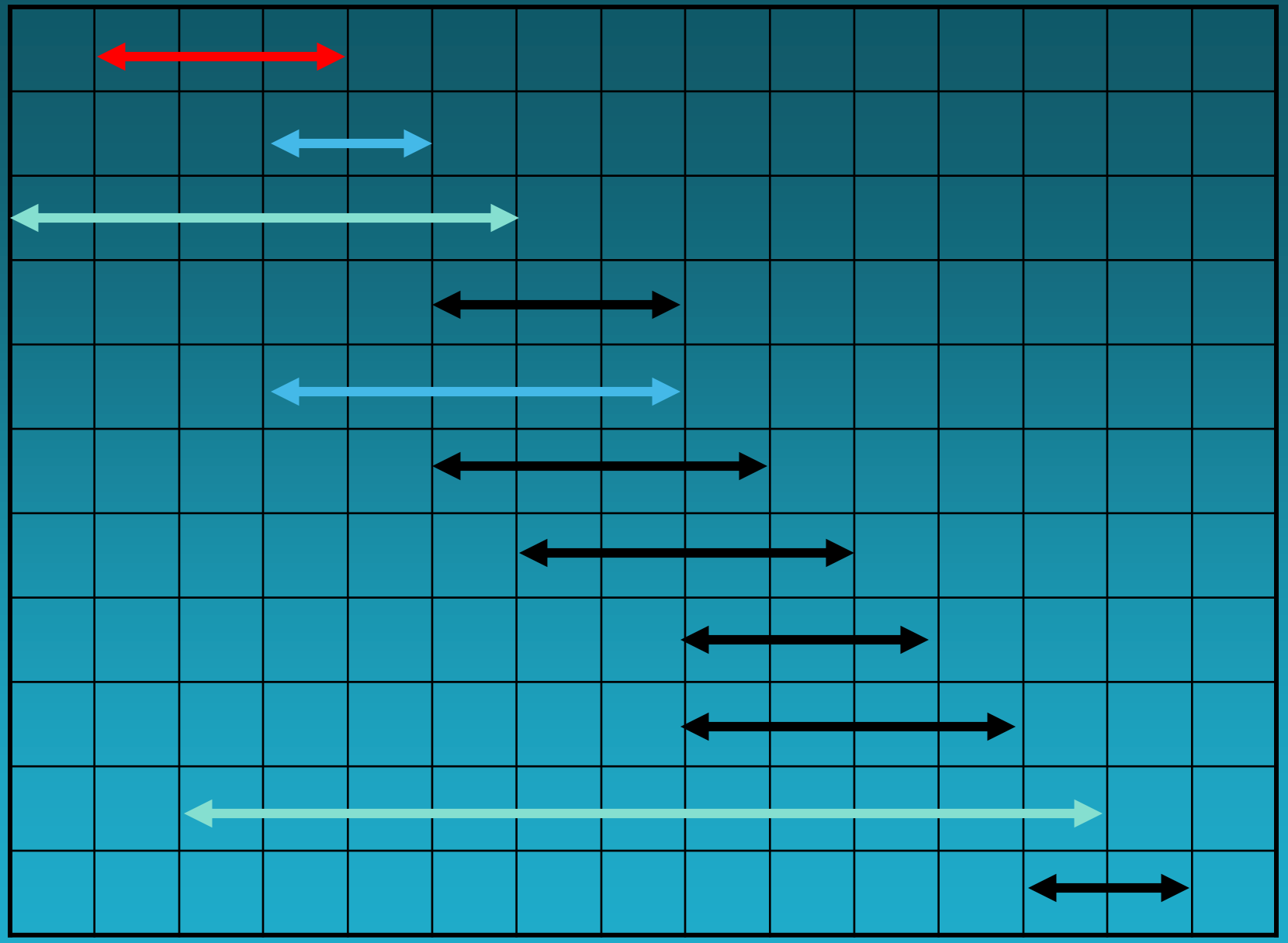


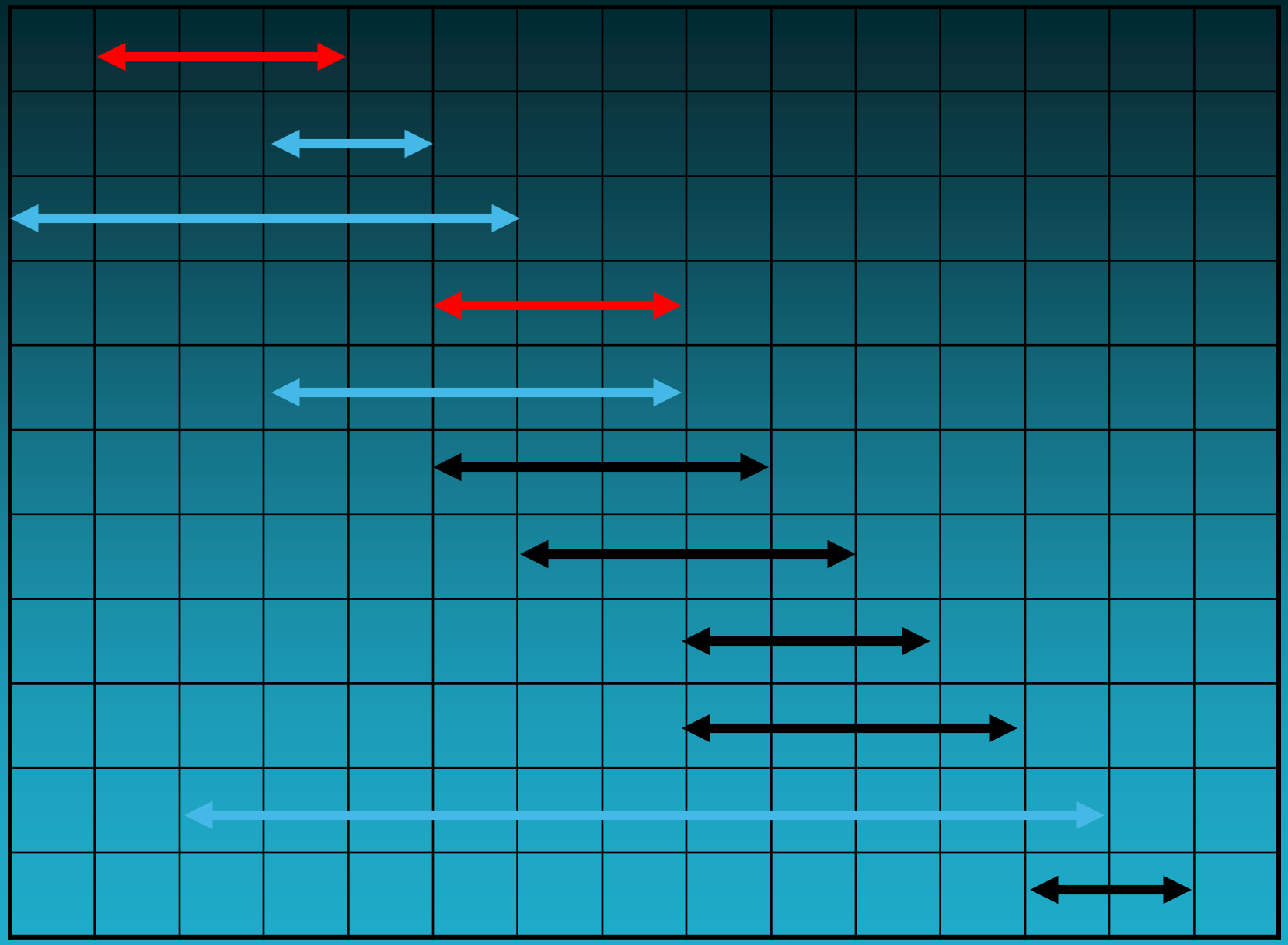


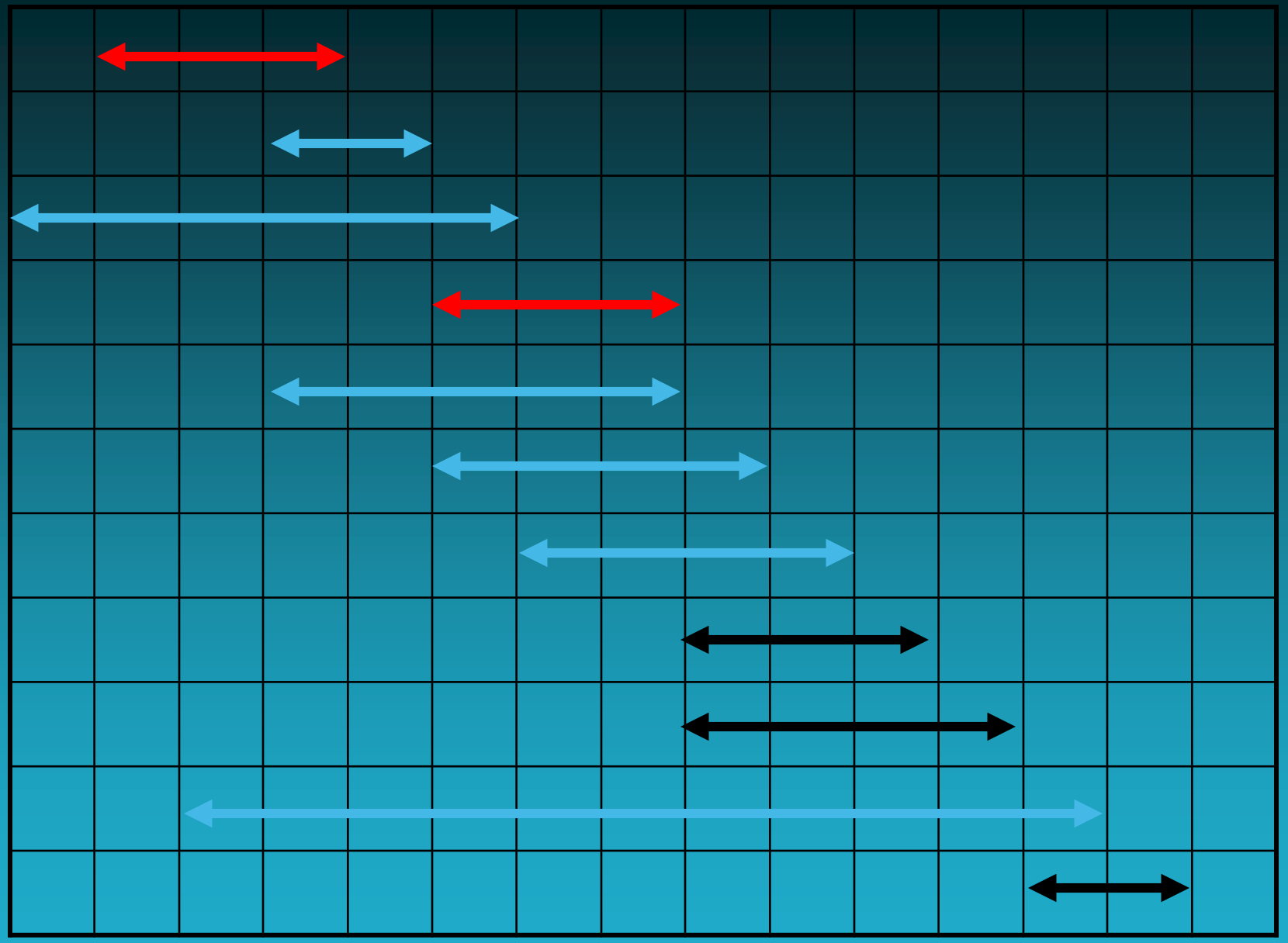
Early Finish Greedy

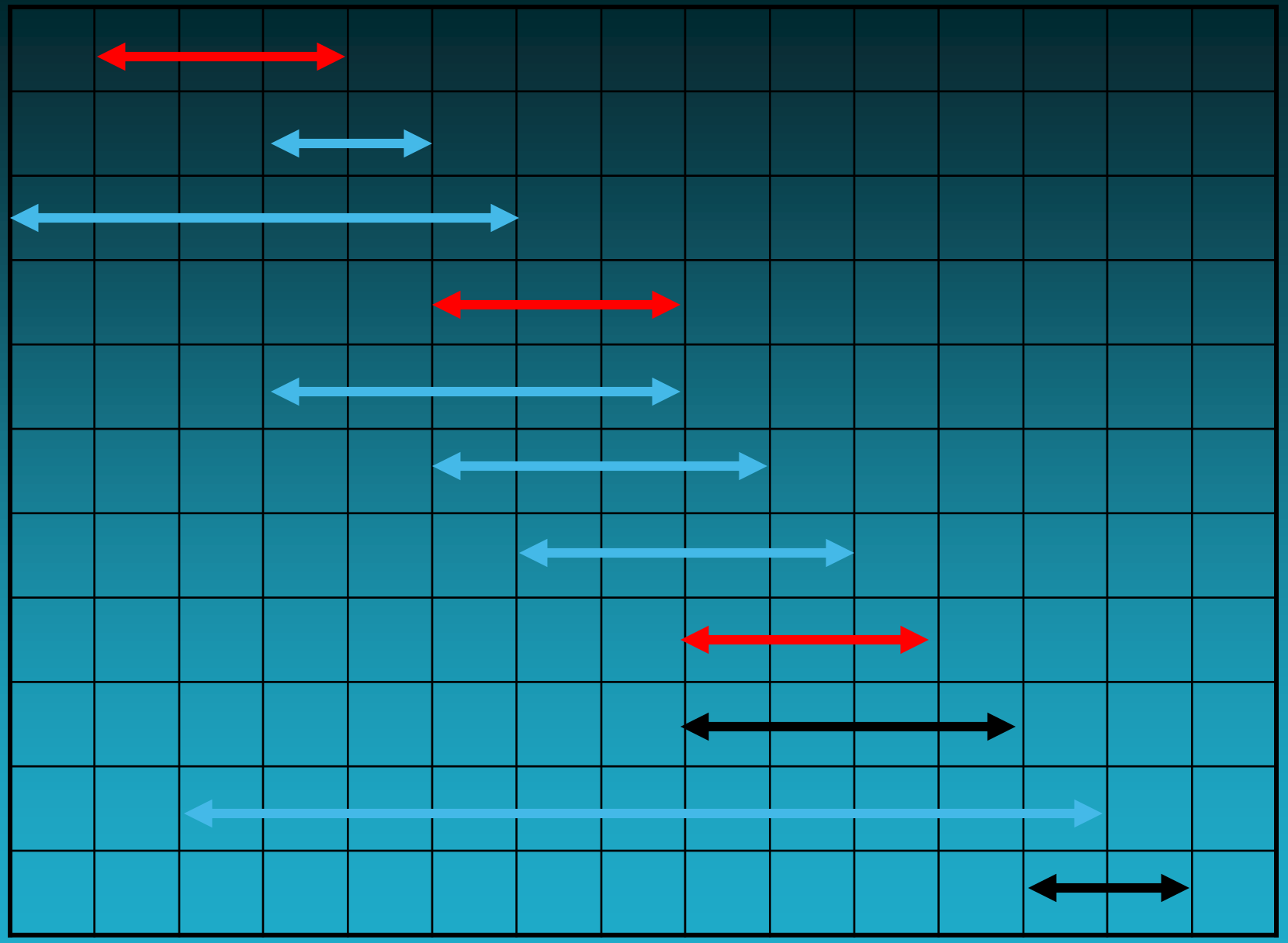
- Select the activity with the earliest finish
- Eliminate the activities that could not be scheduled
- Repeat!

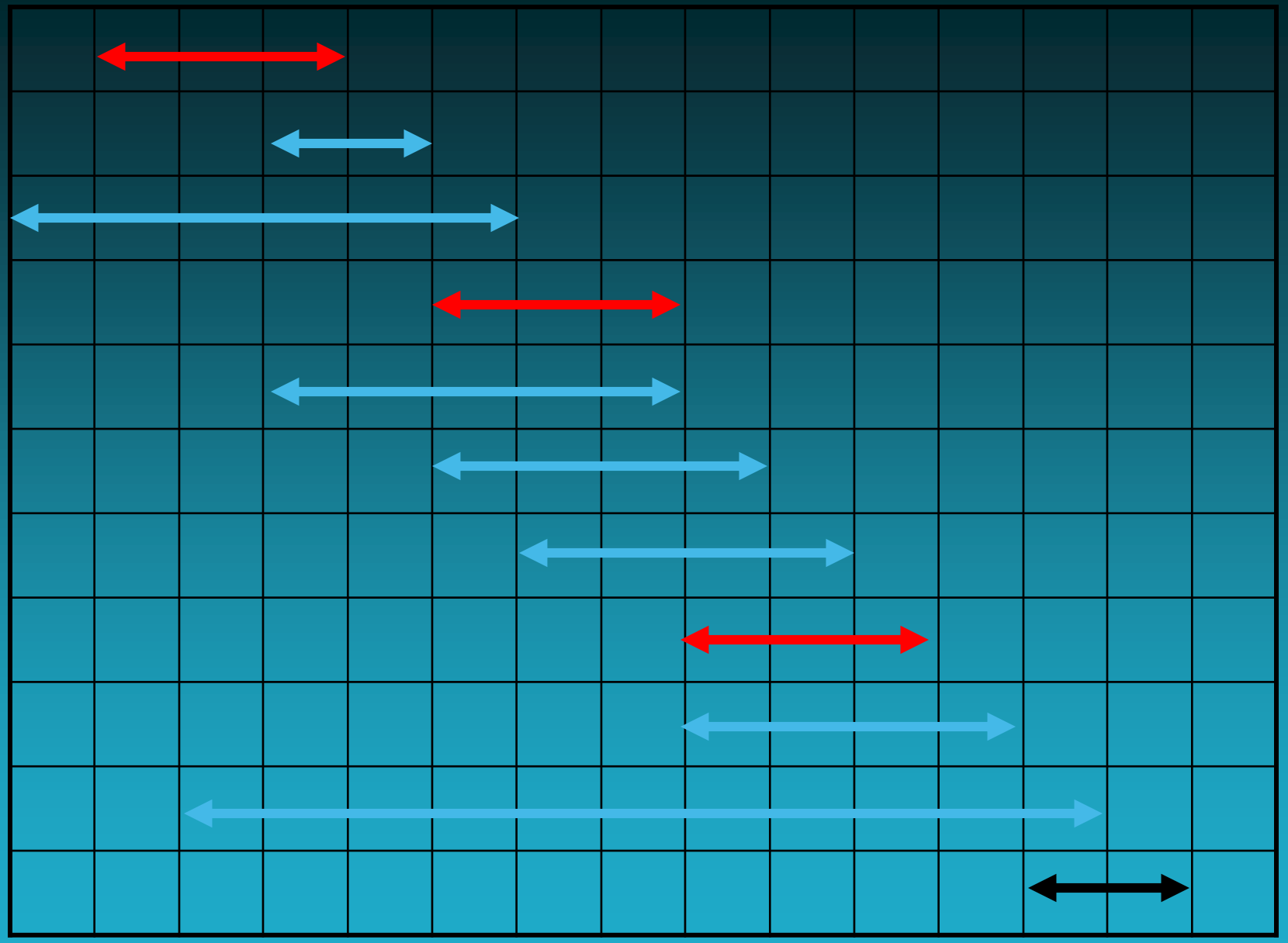


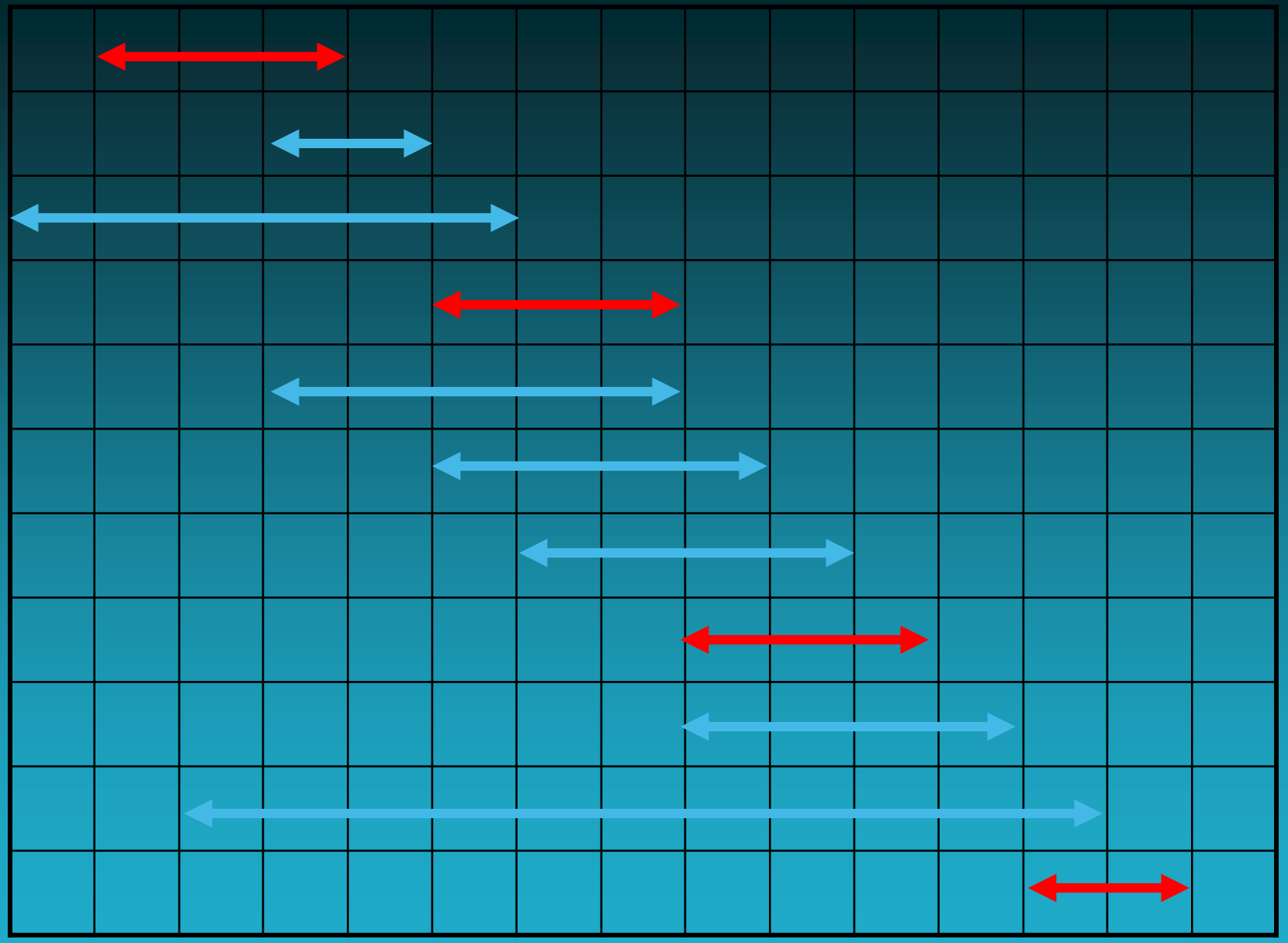


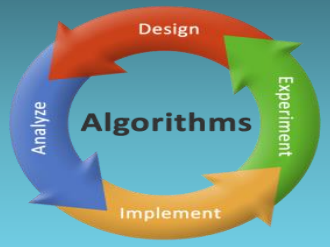












Assuming activities are sorted by finish time

```
GREEDY-ACTIVITY-SELECTOR(s, f)
1   n ← length[s]
2   A ← {a_1}
3   i ← 1
4   for m ← 2 to n
5         do if s_m ≥ f_i
6               then A ← A ∪ {a_m}
7                    i ← m
8   return A
```