make history.

THE UNIVERSITY of ADELAIDE

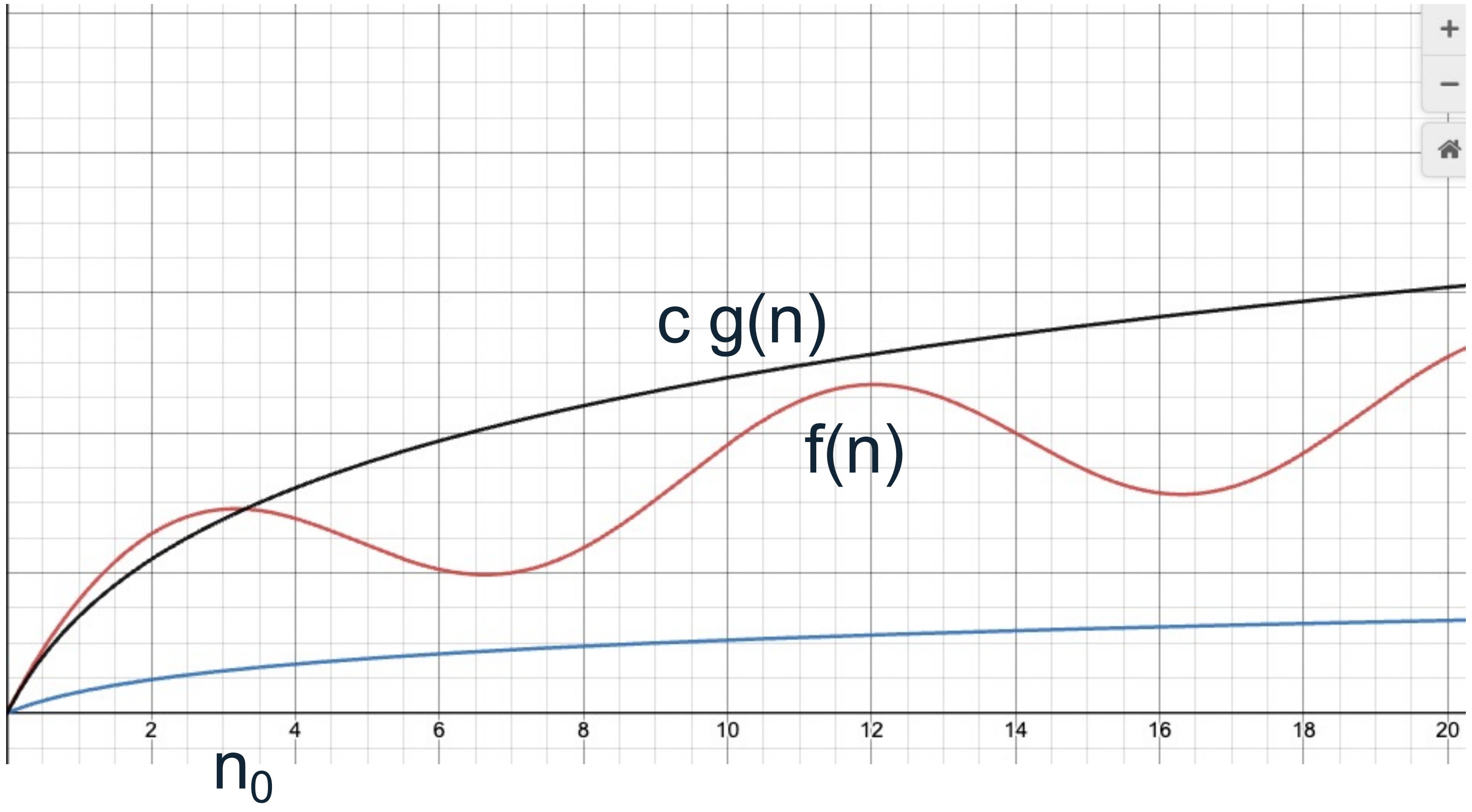
Upper bound: Big O notation

Dr. Anna Kalenkova

Big O – Upper bound

We say that $f(n)$ is in $O(g(n))$ iff (if and only if)

$$\exists c \in R^+, \exists n_0 \in N,\ such\ that\ \forall n \geq n_0 : f(n) \leq cg(n).$$



Exhaustive search. Worst case

	a[0]	a[1]	a[2]	a[3]				a[n-2]	a[n-1]
a	1	3	10	1	…	…	…	9	0

…

```
for(int i=0; i<n; i++) {
      if (a[i] == value) {
            return i; // found it!!!
      }
      …
} …
```

O(n)

Exhaustive search. Best case

	a[0]	a[1]	a[2]	a[3]				a[n-2]	a[n-1]
a	1	3	10	1	…	…	…	9	0

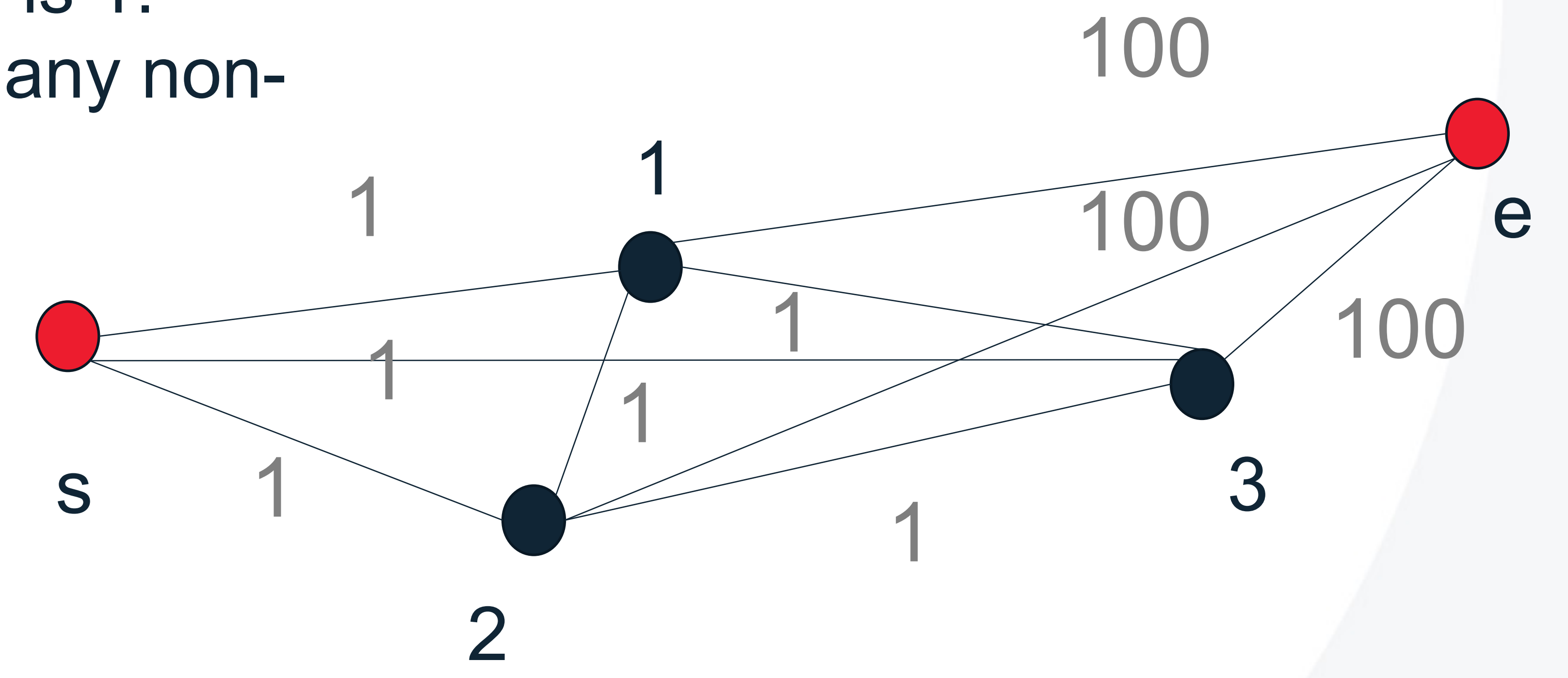
…

```
for(int i=0; i<n; i++) {
    if (a[i] == value) {
        return i; // found it!!!
    }
    …
} …
```

O(1)

Dijkstra’s shortest route. Worst case

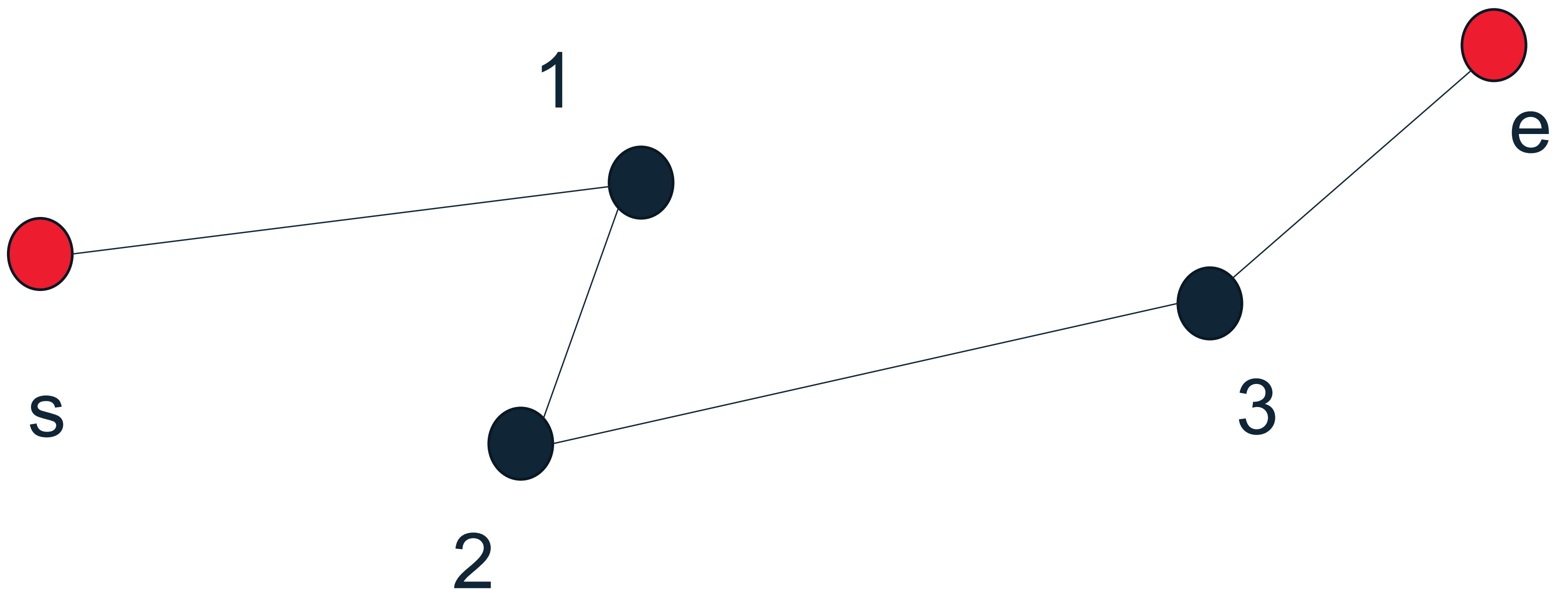
Each point i is connected to each point j.
Distance between any i and any j ($i \neq j$) is 1.
Distance from any i to e is larger than any non-cyclic path inside.



$O(n^2)$

(s,0)	(1,∞)	(2,∞)	(3,∞)	(e, ∞)

Dijkstra’s shortest route. Best case



O(n)