

# Greedy Algorithms

## Lecture 19

Tuesday, November 3, 2020

## 19.1

### Greedy algorithms by example

# Greedy algorithms

Why don't you do right?

- 1 **greedy algorithms**: do locally the right thing...
- 2 ...and they suck.

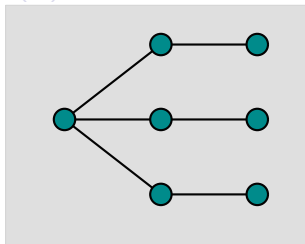
Problem: **VertexCoverMin**

Instance: Vertex Cover!Minimization

Question: A graph  $G$ .

Return the **smallest** subset  $S \subseteq V(G)$ , s.t.  $S$  touches all the edges of  $G$ .

- 3 **GreedyVertexCover**: pick vertex with highest degree, remove, repeat.



# Greedy algorithms

Why don't you do right?

- 1 **greedy algorithms**: do locally the right thing...
- 2 ...and they suck.

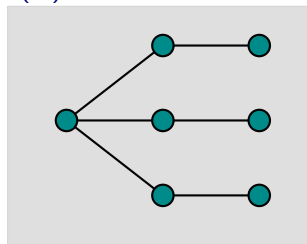
**Problem:** VertexCoverMin

**Instance:** Vertex Cover!Minimization

**Question:** A graph  $G$ .

Return the **smallest** subset  $S \subseteq V(G)$ , s.t.  $S$  touches all the edges of  $G$ .

- 3 **GreedyVertexCover**: pick vertex with highest degree, remove, repeat.



# Greedy algorithms

Why don't you do right?

- 1 **greedy algorithms**: do locally the right thing...
- 2 ...and they suck.

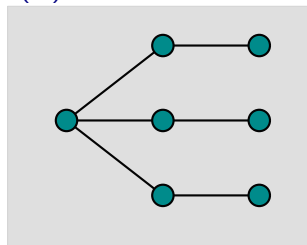
**Problem:** VertexCoverMin

**Instance:** Vertex Cover!Minimization

**Question:** A graph  $G$ .

Return the **smallest** subset  $S \subseteq V(G)$ , s.t.  $S$  touches all the edges of  $G$ .

- 3 **GreedyVertexCover**: pick vertex with highest degree, remove, repeat.



# Greedy algorithms

Why don't you do right?

- 1 **greedy algorithms**: do locally the right thing...
- 2 ...and they suck.

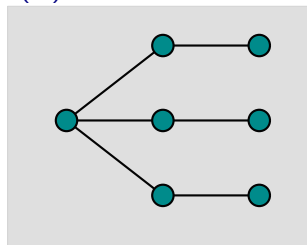
**Problem:** VertexCoverMin

**Instance:** Vertex Cover!Minimization

**Question:** A graph  $G$ .

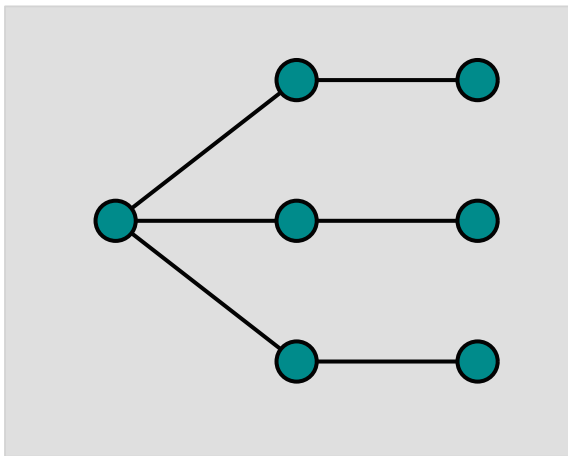
Return the **smallest** subset  $S \subseteq V(G)$ , s.t.  $S$  touches all the edges of  $G$ .

- 3 **GreedyVertexCover**: pick vertex with highest degree, remove, repeat.



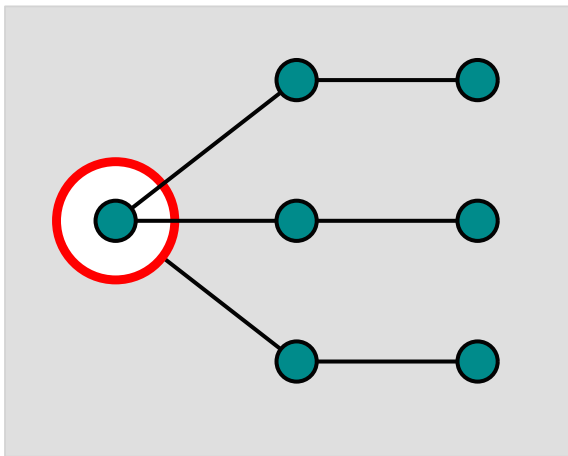
# Greedy algorithms

GreedyVertexCover in action...



# Greedy algorithms

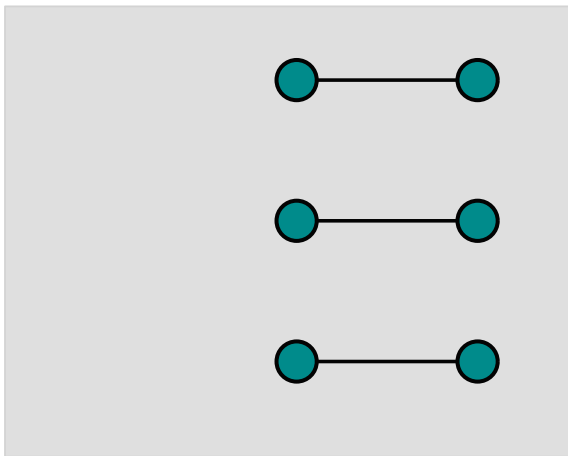
GreedyVertexCover in action...





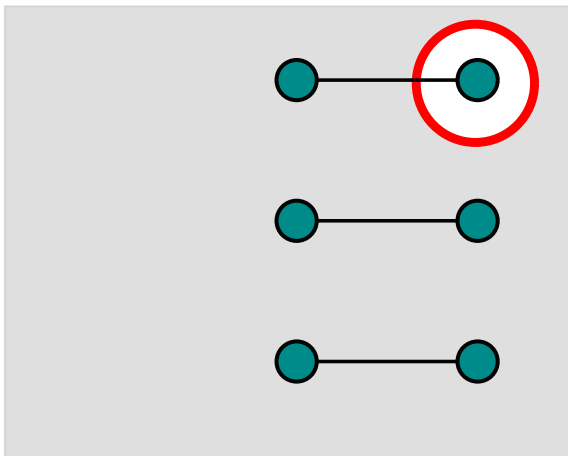
# Greedy algorithms

GreedyVertexCover in action...



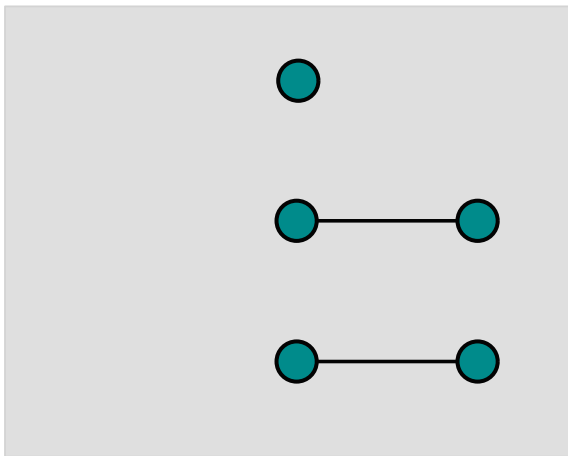
# Greedy algorithms

GreedyVertexCover in action...



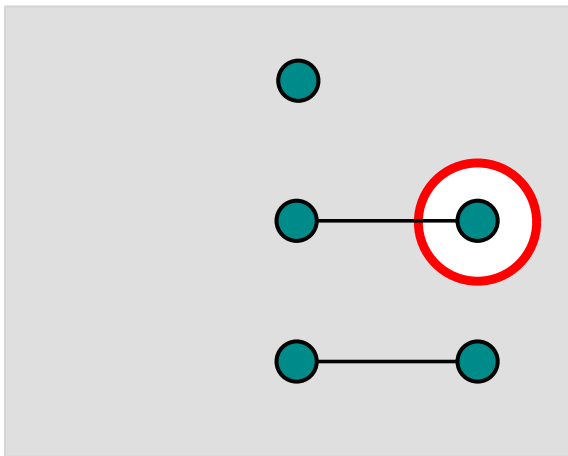
# Greedy algorithms

GreedyVertexCover in action...



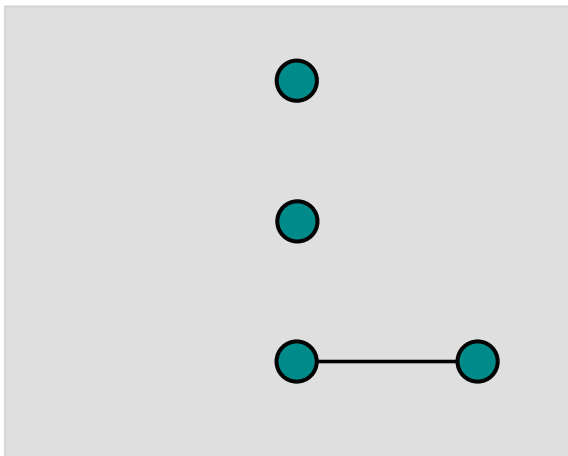
# Greedy algorithms

GreedyVertexCover in action...



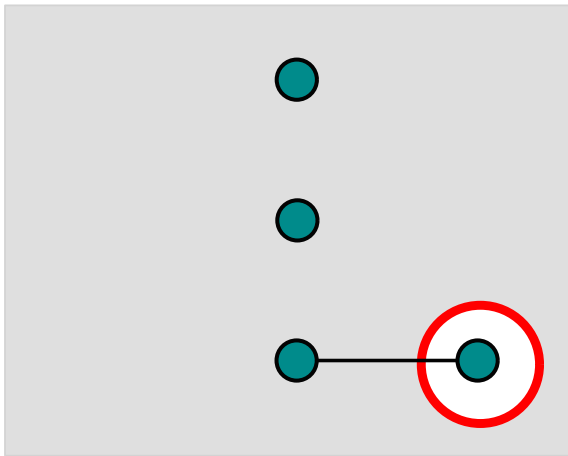
# Greedy algorithms

GreedyVertexCover in action...



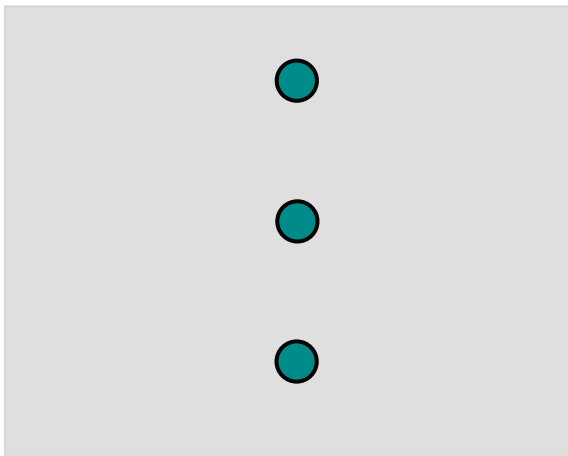
# Greedy algorithms

GreedyVertexCover in action...



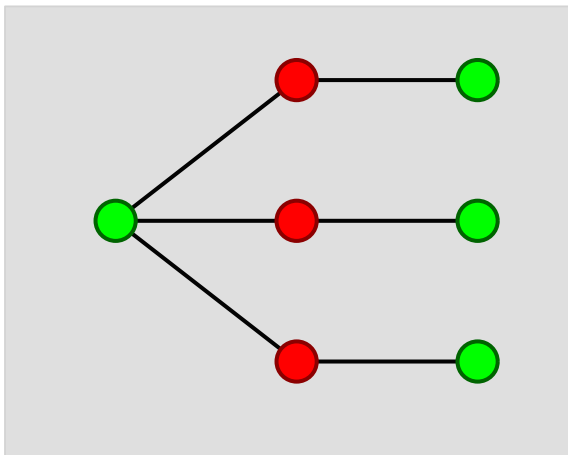
# Greedy algorithms

GreedyVertexCover in action...



# Greedy algorithms

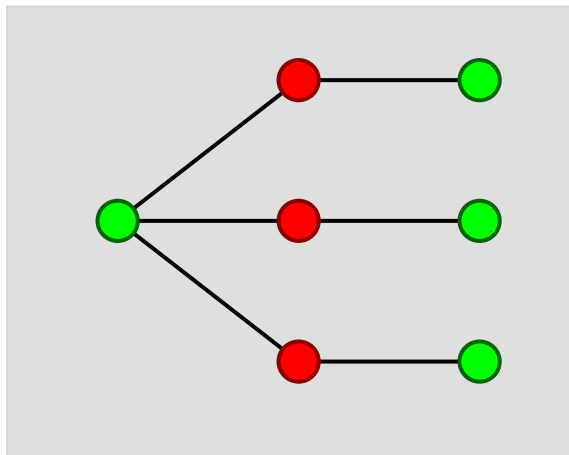
GreedyVertexCover in action...





# Greedy algorithms

GreedyVertexCover in action...

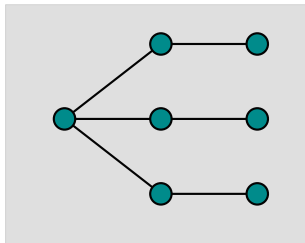


## Observation 19.1.

**GreedyVertexCover** returns 4 vertices, but *opt* is 3 vertices.

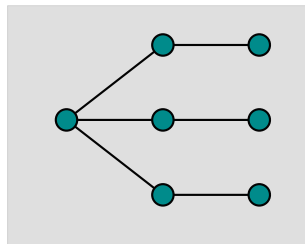
## Back to GreedyVertexCover

- 1 **GreedyVertexCover**: pick vertex with highest degree, remove, repeat.
- 2 Returns 4, but opt is 3!
- 3 Can **not** be better than a  $4/3$ -approximation algorithm.
- 4 Actually it is much worse!



## Back to GreedyVertexCover

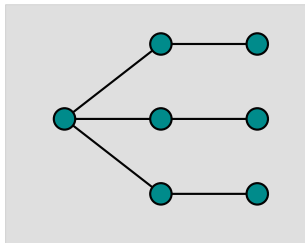
- 1 **GreedyVertexCover**: pick vertex with highest degree, remove, repeat.
- 2 Returns 4, but opt is 3!



- 3 Can **not** be better than a  $4/3$ -approximation algorithm.
- 4 Actually it is much worse!

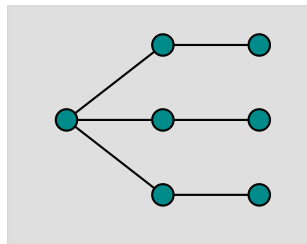
## Back to GreedyVertexCover

- 1 **GreedyVertexCover**: pick vertex with highest degree, remove, repeat.
- 2 Returns 4, but opt is 3!
- 3 Can **not** be better than a  $4/3$ -approximation algorithm.
- 4 Actually it is much worse!



## Back to GreedyVertexCover

- 1 **GreedyVertexCover**: pick vertex with highest degree, remove, repeat.
- 2 Returns 4, but opt is 3!
- 3 Can **not** be better than a  $4/3$ -approximation algorithm.
- 4 Actually it is much worse!



# Greedy Vertex Cover

## Theorem 19.2.

*There is a graph over  $n$  vertices, such that the smallest Vertex Cover has  $k$  vertices, but the greedy algorithm outputs a vertex cover of size  $\Theta(k \log n)$  approximation.*

Proof: Outside the scope of this class...

...left as a **hard** exercise to the interested reader.

---

Vertex Cover is NP-Hard: Believe it requires exponential time to solve exactly.

## Greedy Vertex Cover

### Theorem 19.2.

*There is a graph over  $n$  vertices, such that the smallest Vertex Cover has  $k$  vertices, but the greedy algorithm outputs a vertex cover of size  $\Theta(k \log n)$  approximation.*

Proof: Outside the scope of this class...

...left as a **hard** exercise to the interested reader.

---

**Vertex Cover** is **NP-Hard**: Believe it requires exponential time to solve exactly.

**THE END**

...

**(for now)**