```
VC-2-approximation(G):
    while uncovered edge exists in G
        select both vertices from uncovered edge
```

f is $O(g)$ $\approx$ $f \le g$

We say that an algorithm is an α-approximation for a problem if ALG ≤ $\underset{2}{\alpha}$ OPT for all inputs.

Answer the following questions with True or False.

1. VC-2-approximation always finds a vertex cover for input G.

T

2. VC-2-approximation always finds the optimal (smallest size) vertex cover for input G.

F

3. VC-2-approximation is a 3-approximation for the smallest vertex cover problem.

True

4. VC-2-approximation is a 1.5-approximation for the smallest vertex cover problem.

$ALG \nleq 1.5\, OPT$

$2 \nleq 1.5 \cdot 1$

$ALG = 2$

$OPT = 1$

Set Cover

$\mathcal{U}$

Set Cover: Given a set of elements (the universe), and sets containing those elements, find the smallest number of sets so that every element of the universe is included.

Set Cover

Set Cover: Given a set of elements (the universe), and sets containing those elements, find the smallest number of sets so that every element of the universe is included.

Example:

$$U = \{1, 4, 7, 8, 10\}$$

$$S = \{\{1, 7, 8\}, \{1, 4, 7\}, \{7, 8\}, \{4, 8, 10\}\}$$

Set Cover

Set Cover: Given a set of elements (the universe), and sets containing those elements, find the smallest number of sets so that every element of the universe is included.

Example:

$$U = \{1, 4, 7, 8, 10\}$$

$$S = \{\{1, 7, 8\}, \{1, 4, 7\}, \{7, 8\}, \{4, 8, 10\}\}$$

$$\{\{1, 7, 8\}, \{4, 8, 10\}\} \checkmark$$

Set Cover

Set Cover: Given a set of elements (the universe), and sets containing those elements, find the smallest number of sets so that every element of the universe is included.

Example:

$$U = \{1, 4, 7, 8, 10\}$$

$$S = \{\{1, 7, 8\}, \underline{\{1, 4, 7\}}, \underline{\{7, 8\}}, \{4, 8, 10\}\}$$

$$\{\{1, 7, 8\}, \{4, 8, 10\}\} \qquad \{\{1, 4, 7\}, \{7, 8\}\} \quad \times$$

Set Cover

Set Cover: Given a set of elements (the universe), and sets containing those elements, find the smallest number of sets so that every element of the universe is included.

Example:

$$U = \{1, 4, 7, 8, 10\}$$

$$S = \{\{1, 7, 8\}, \{1, 4, 7\}, \{7, 8\}, \{4, 8, 10\}\}$$

$$\{\{1, 7, 8\}, \{4, 8, 10\}\} \qquad \{\{1, 4, 7\}, \{7, 8\}\}$$

$$\{\{1, 4, 7, 8, 10\}\}$$

Set Cover

Set Cover: Given a set of elements (the universe), and sets containing those elements, find the smallest number of sets so that every element of the universe is included.

Example:

$$U = \{1, 4, 7, 8, 10\}$$

$$S = \{\{1, 7, 8\}, \{1, 4, 7\}, \{7, 8\}, \{4, 8, 10\}\}$$

$$\{\{1, 7, 8\}, \{4, 8, 10\}\} \qquad \{\{1, 4, 7\}, \{7, 8\}\}$$

(of size k)

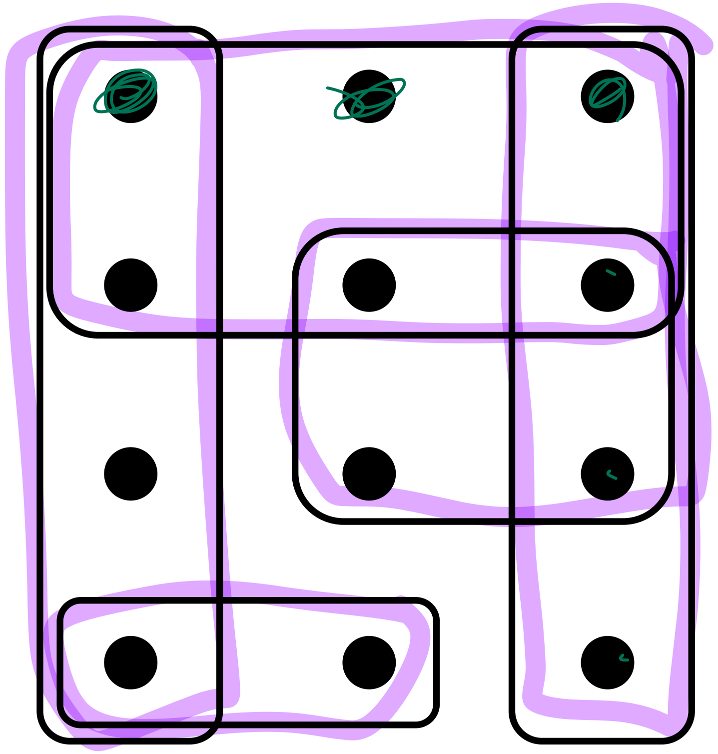
is set cover in NP?

$$\{\{1, 4, 7, 8, 10\}\}$$

Set Cover

Set Cover: Given a set of elements (the universe), and sets containing those elements, find the smallest number of sets so that every element of the universe is included.

Example:



$|U| = 12$

$|S| = 5$

what is smallest set cover?

Set Cover

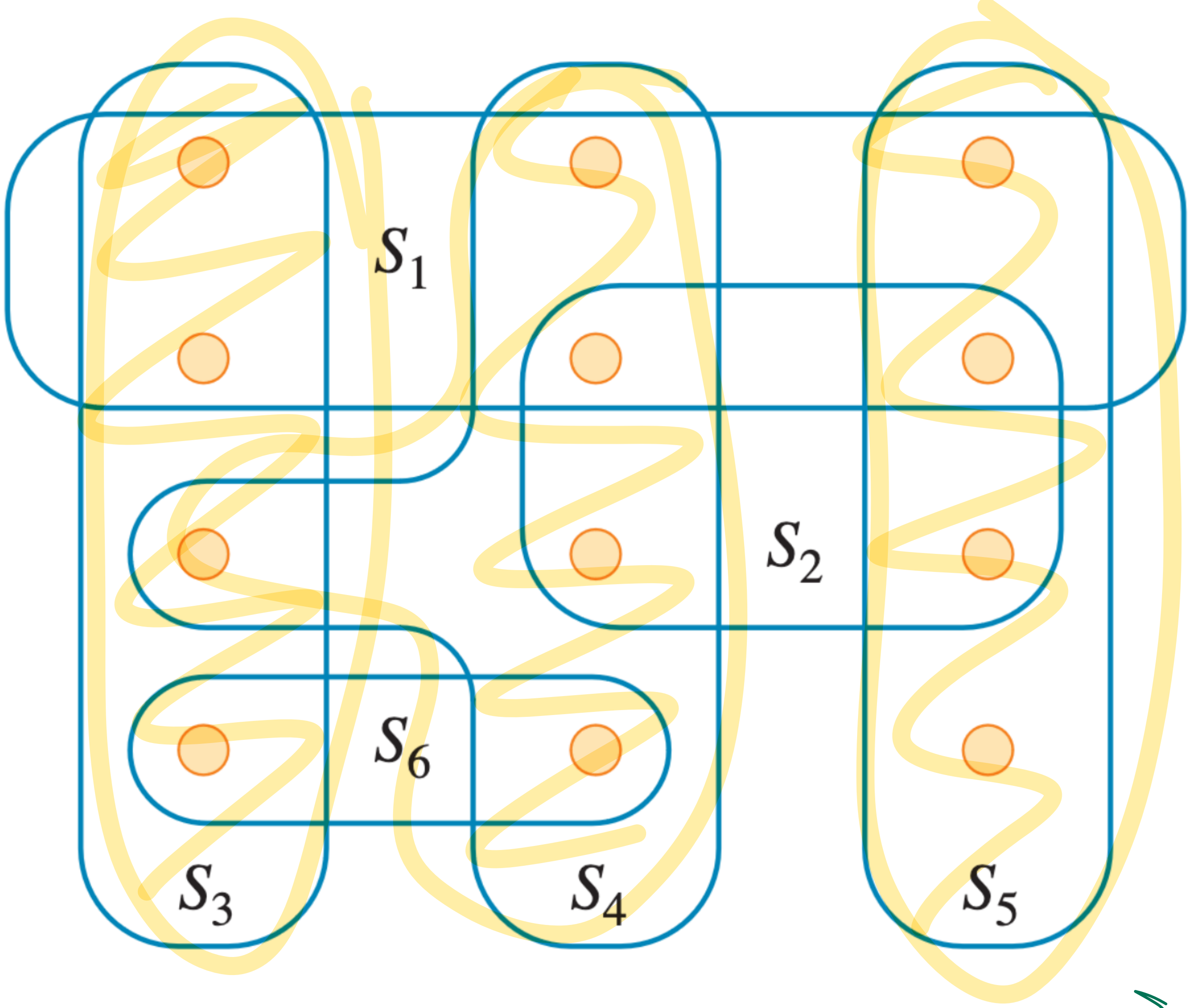
NP-hard

Set Cover: Given a set of elements (the universe), and sets containing those elements, find the smallest number of sets so that every element of the universe is included.

Greedy Algorithm:

```
while element of universe not included
  select Si with largest number of excluded elements.
```

```
while element of universe not included
   select Si with largest number of excluded elements.
```



S_1
S_4
S_5
S_3

give sets chosen by algorithm. are they an optimal set cover? → no, can be done w/ 3

Set Cover

Set Cover: Given a set of elements (the universe), and sets containing those elements, find the smallest number of sets so that every element of the universe is included.

Greedy Algorithm:

```
while element of universe not included
  select S_i with largest number of excluded elements.
```

1. Valid?
2. Polynomial Time?
3. Performance?

Set Cover

Set Cover: Given a set of elements (the universe), and sets containing those elements, find the smallest number of sets so that every element of the universe is included.

Greedy Algorithm:

```
while element of universe not included
  select Si with largest number of excluded elements.
```

1. Valid. Every element of universe will be included.
2. Polynomial Time. $O(|S|^2|U|)$.
3. Performance?

Greedy Set Cover – Performance

```
while element of universe not included
   select Si with largest number of excluded elements.
```

Greedy Set Cover – Performance

```
while element of universe not included
   select Si with largest number of excluded elements.
```

Goal: ALG ≤ α OPT

Greedy Set Cover – Performance

ALG = ?

OPT = ?

```
while element of universe not included
   select Sᵢ with largest number of excluded elements.
```

$$ALG \leq \alpha \, OPT$$

Greedy Set Cover – Performance

ALG = # sets selected by the algorithm to cover all n elements.

OPT = # sets in an optimal solution to cover all n elements.

```
while element of universe not included
   select Sᵢ with largest number of excluded elements.
```

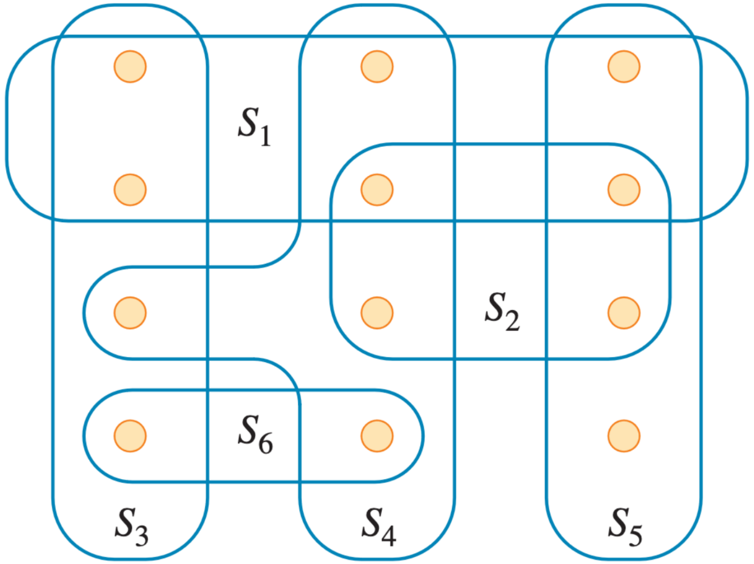
$$\text{ALG} \leq \alpha\ \text{OPT}$$

Greedy Set Cover – Performance

ALG = # sets selected by the algorithm to cover all n elements.

OPT = # sets in an optimal solution to cover all n elements.

```
while element of universe not included
   select Si with largest number of excluded elements.
```



4 3

What are ALG and OPT for our example?

$$ALG \le \alpha\ OPT$$

Greedy Set Cover – Performance

ALG = # sets selected by the algorithm to cover all n elements.

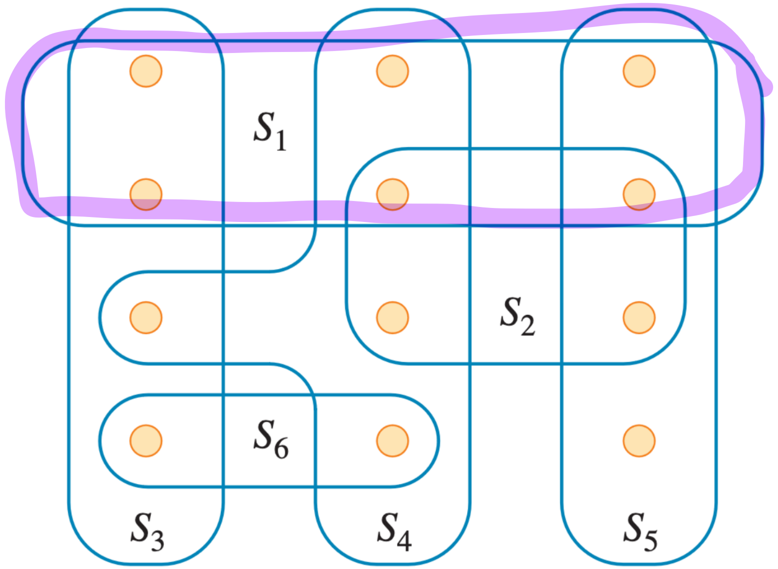
OPT = # sets in an optimal solution to cover all n elements.

```
while element of universe not included
   select Si with largest number of excluded elements.
```

Suppose the universe contains n elements.

Let S be the set first set selected by the greedy algorithm. Can you say anything about |S|?

Hint: in terms of n and OPT?



$|S|$

$OPT \geq \frac{n}{|S|}$

Goal: ALG ≤ α OPT

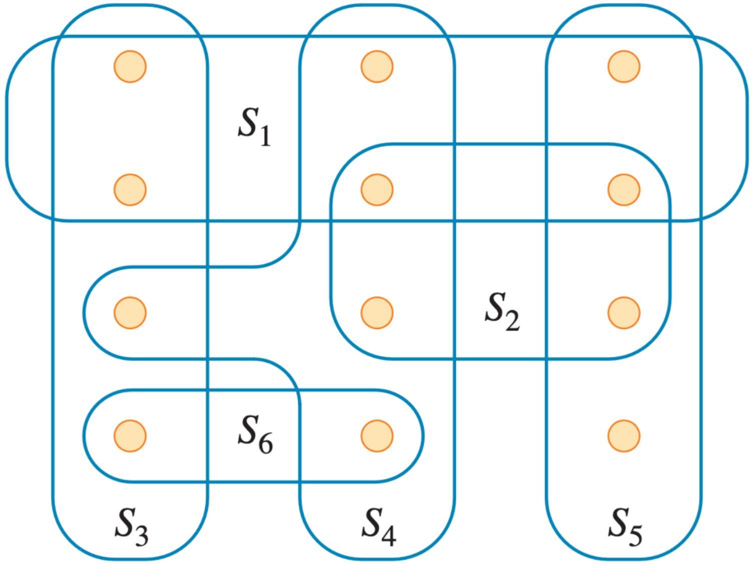
Greedy Set Cover – Performance

ALG = # sets selected by the algorithm to cover all n elements.

OPT = # sets in an optimal solution to cover all n elements.

```
while element of universe not included
   select Si with largest number of excluded elements.
```

Suppose the universe contains n elements.



Let S be the set first set selected by the greedy algorithm. Can you say anything about |S|?

after first iteration,
excluded $\leq$
$n - \frac{n}{OPT}$

$$|S| \geq \frac{n}{OPT}$$

$$\text{ALG} \leq \alpha \text{ OPT}$$

Set Cover – Performance

ALG = # sets selected by the algorithm to cover all n elements.

OPT = # sets in an optimal solution to cover all n elements.

Suppose the universe contains n elements.

Before the t^{th} iteration, some remaining set has at least $\frac{n_{t-1}}{\mathrm{OPT}}$ uncovered elements and the number of elements remaining after the t^{th} iteration is:

$$n_t \le n_{t-1} - \frac{n_{t-1}}{\mathrm{OPT}} = n_{t-1}\left(1 - \frac{1}{\mathrm{OPT}}\right) \le n\left(1 - \frac{1}{\mathrm{OPT}}\right)^t$$

Accepting that $1 - x < e^{-x}$ for all $x \ne 0$,

$$n_t \le n\left(1 - \frac{1}{\mathrm{OPT}}\right)^t < n\left(e^{-\frac{1}{\mathrm{OPT}}}\right)^t = ne^{-\frac{t}{\mathrm{OPT}}}$$

If $t = \mathrm{OPT} \ln n$, $n_t < ne^{-\frac{\mathrm{OPT} \ln n}{\mathrm{OPT}}} = 1$, which means that no elements remain.

So, the universe is covered after at most $t = \mathrm{OPT} \ln n$ iterations.

$$\Longrightarrow \mathrm{ALG} \le \ln n \, \mathrm{OPT}$$

$\alpha = \ln(n)$

Set Cover – Inapproximability

It turns out that Set Cover cannot be approximated within the bound of $\left(1 - o(1)\right) \ln n$, unless P = NP.