PROBLEM SOLVING

BACK

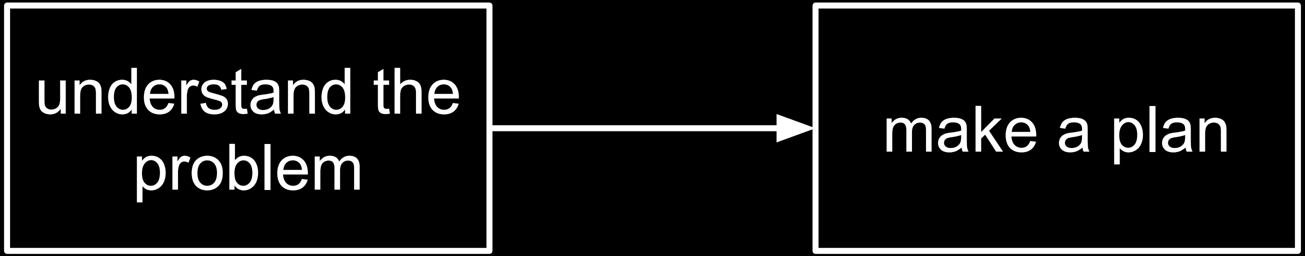
Pólya's approach to problem-solving

image source: http://doi.org/10.3932/ethz-a-000099441

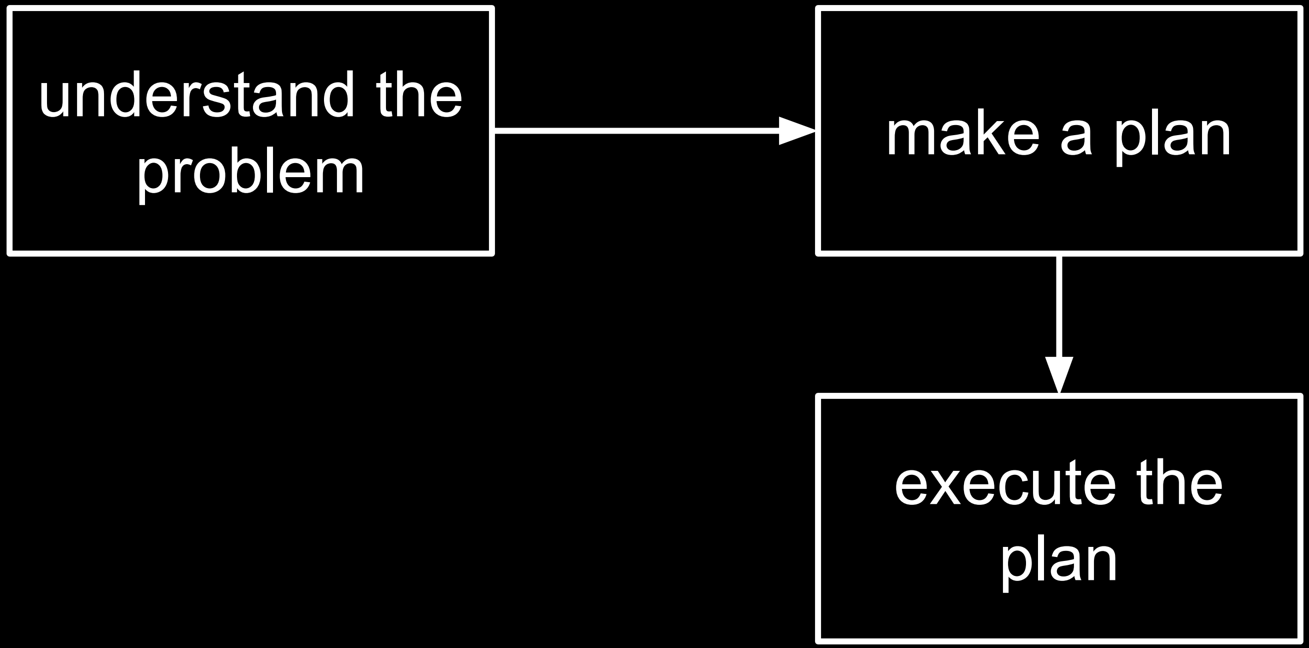
Pólya's approach to problem-solving

understand the problem

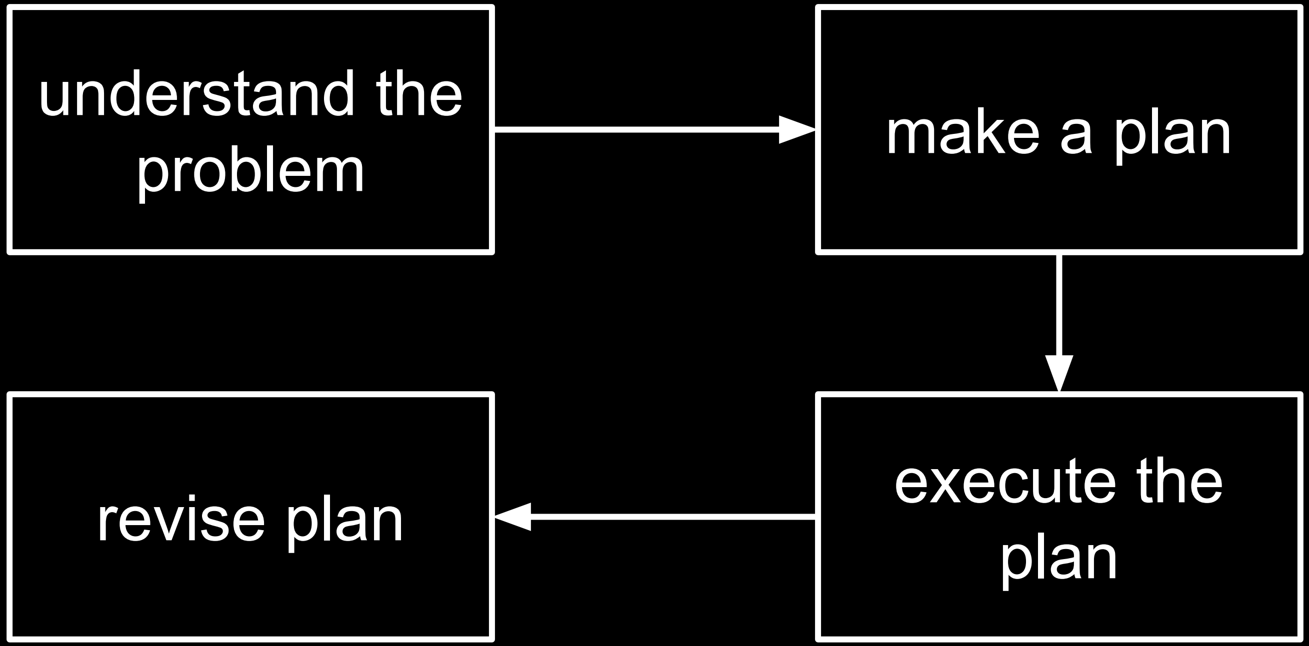
Pólya's approach to problem-solving



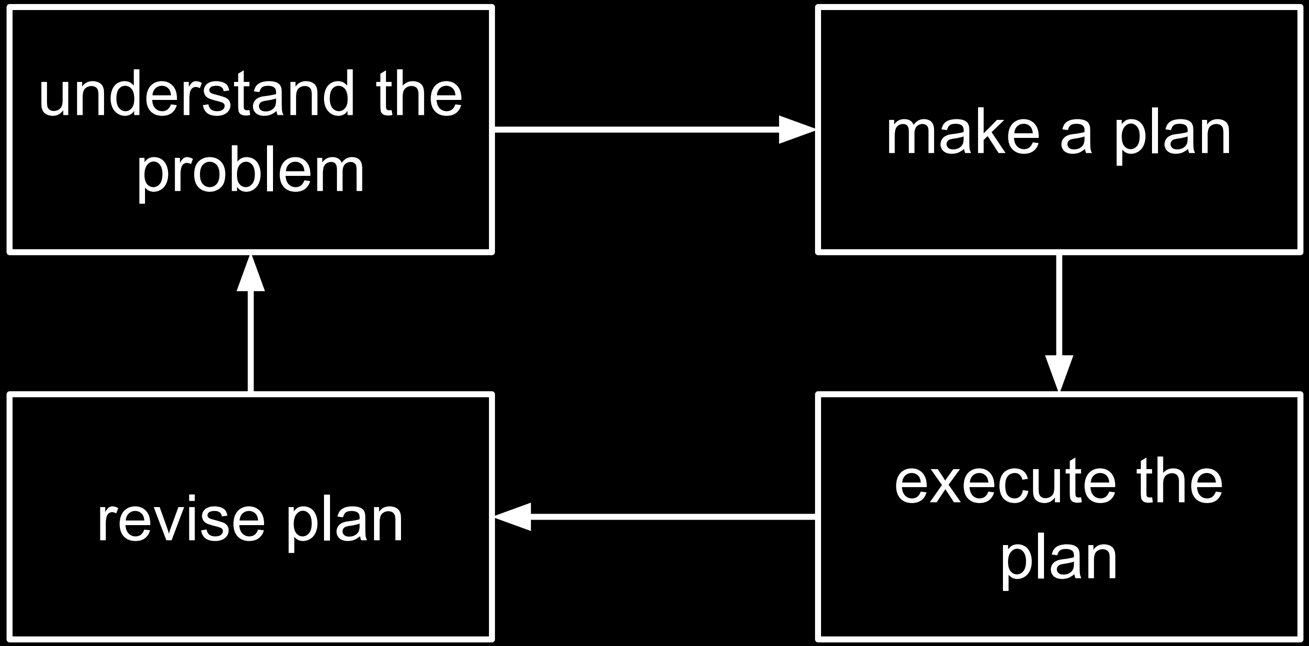
Pólya's approach to problem-solving



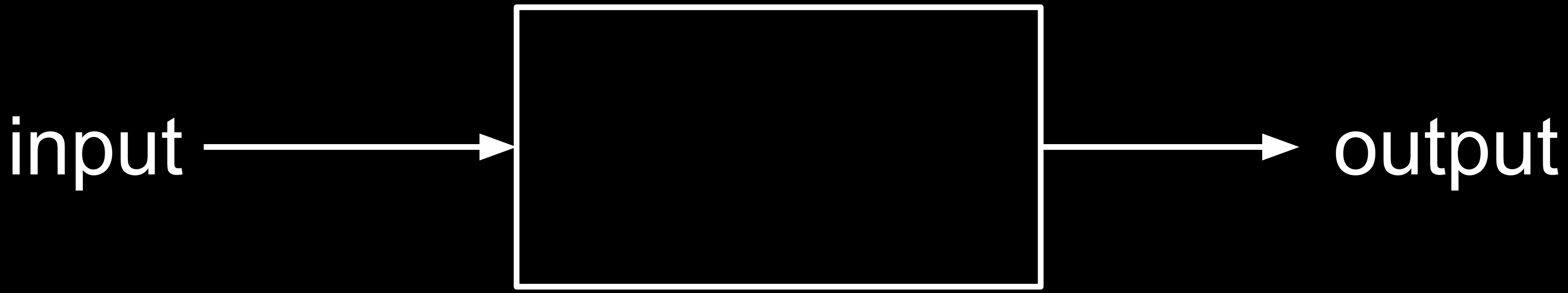
Pólya's approach to problem-solving



Pólya's approach to problem-solving

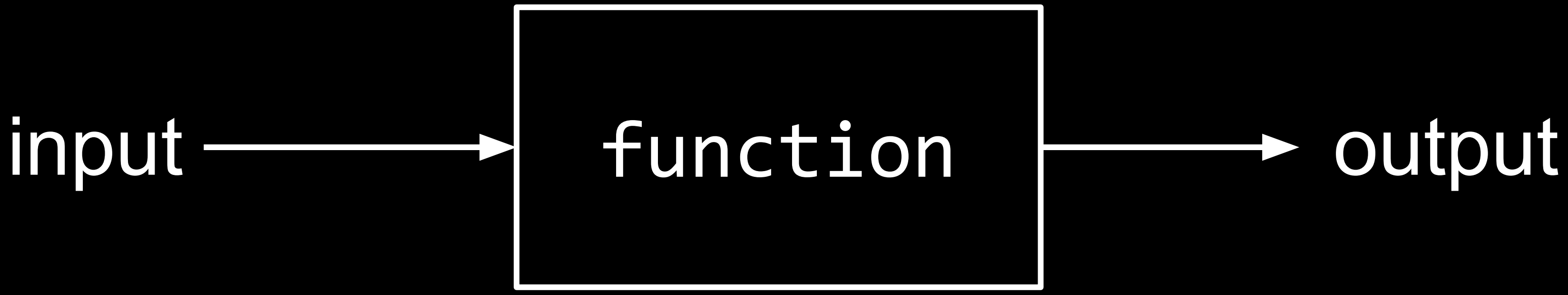


a model to represent problems

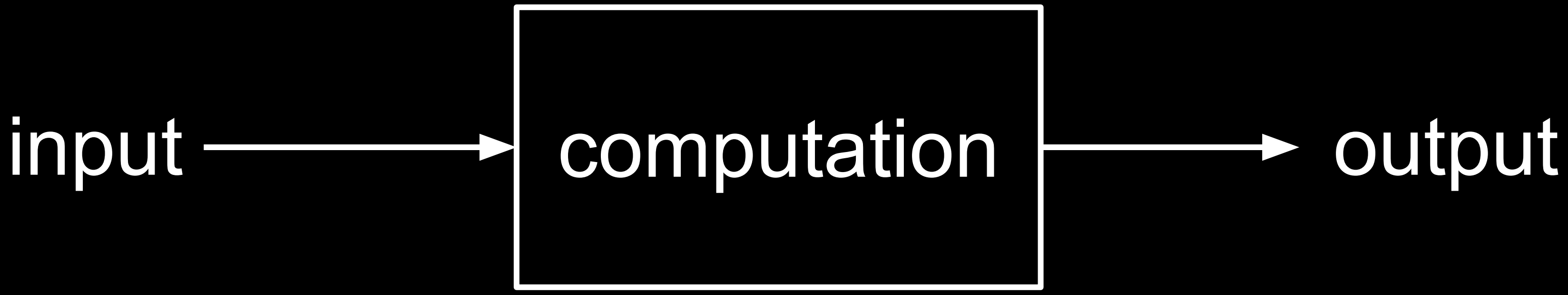


a model to represent problems

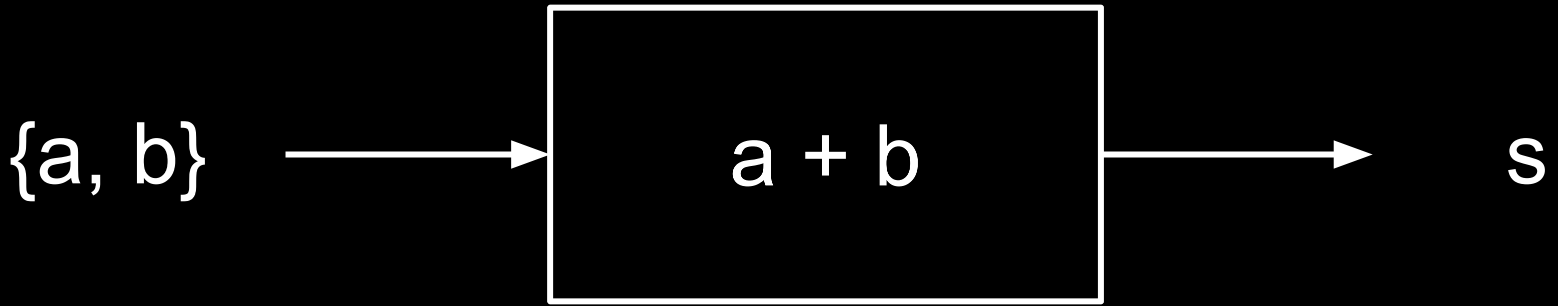
a model to represent problems

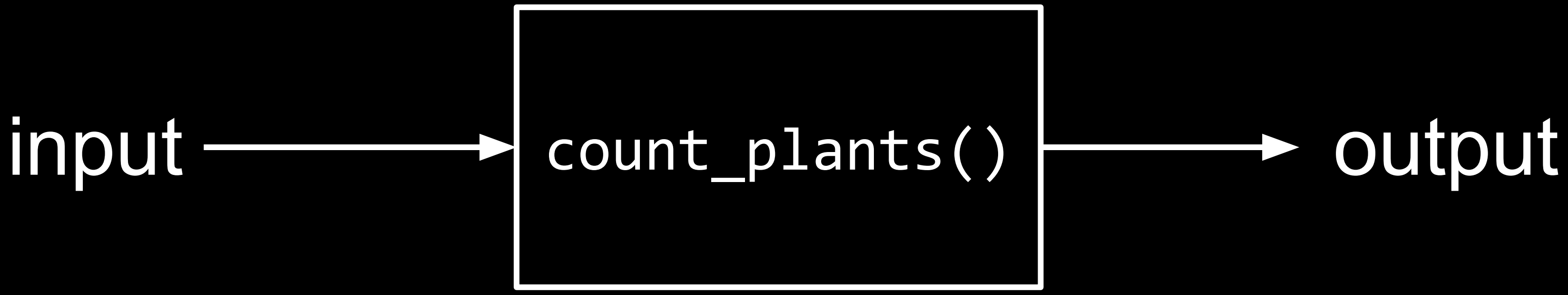


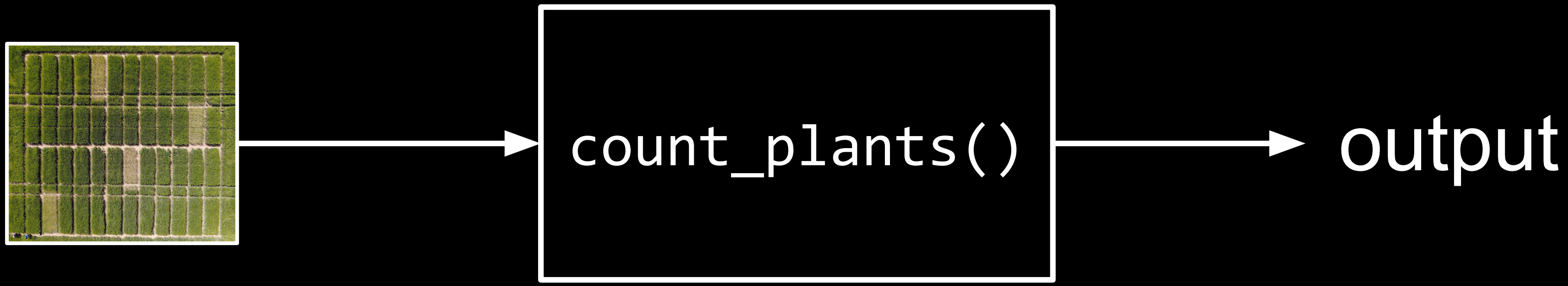
a model to represent problems

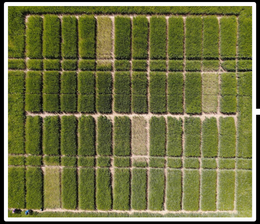


the rainbow experiment as an input - processing - output - problem


{a, b}
a + b
s

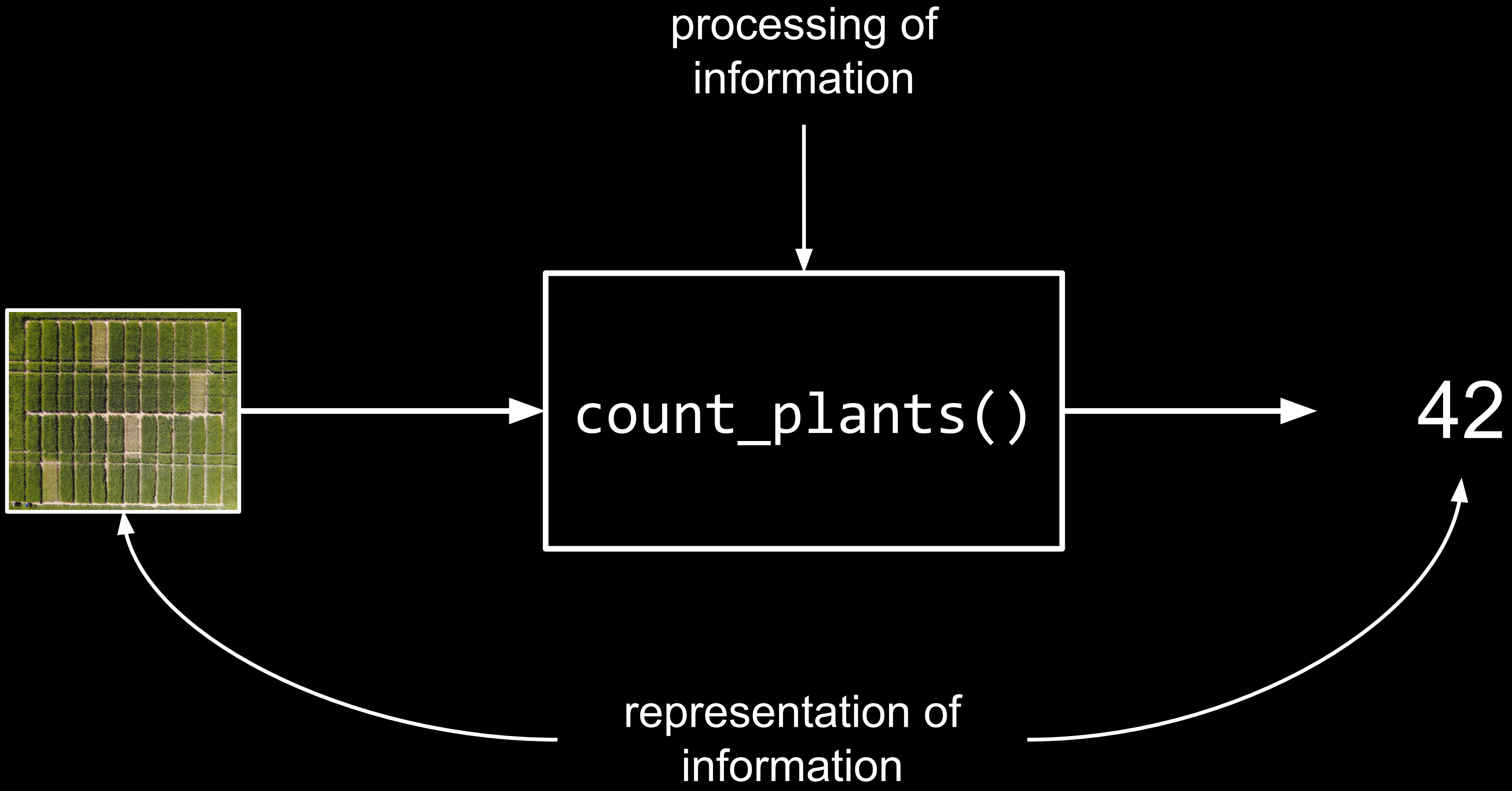

input
count_plants()
output


count_plants()
output

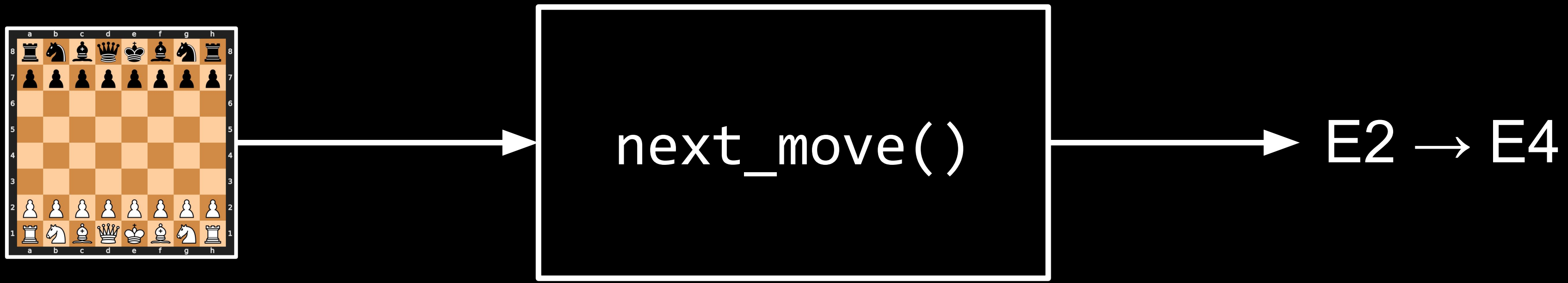


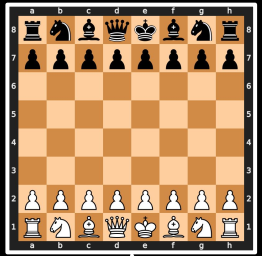
count_plants()

42



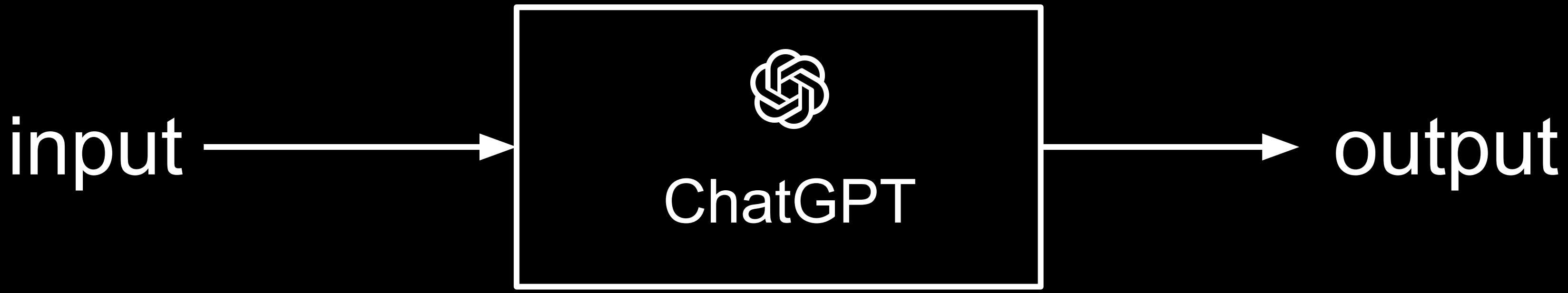
processing of information
count_plants()
42
representation of information


next_move()
E2 → E4



1: 0R	9: 0P		57: 1R
2: 0N	10: 0P		58: 1N
3: 0B	11: 0P		59: 1B
4: 0K	12: 0P		60: 1K
5: 0Q	13: 0P	…	61: 1Q
6: 0B	14: 0P		62: 1B
7: 0N	15: 0P		63: 1N
8: 0R	16: 0P		64: 1R

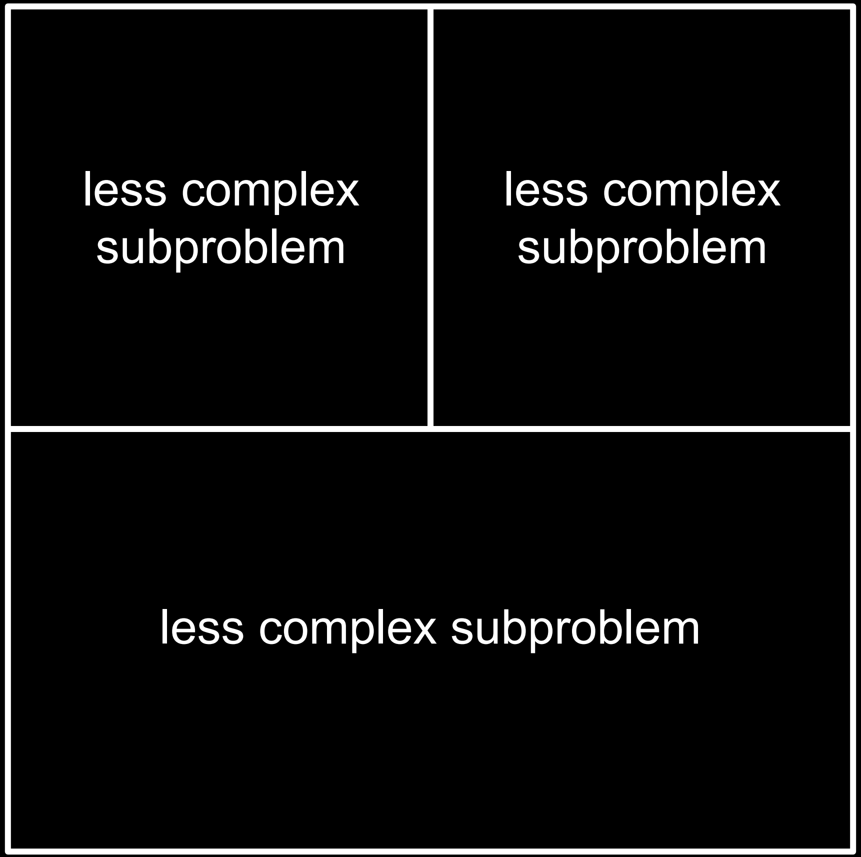
representation of information

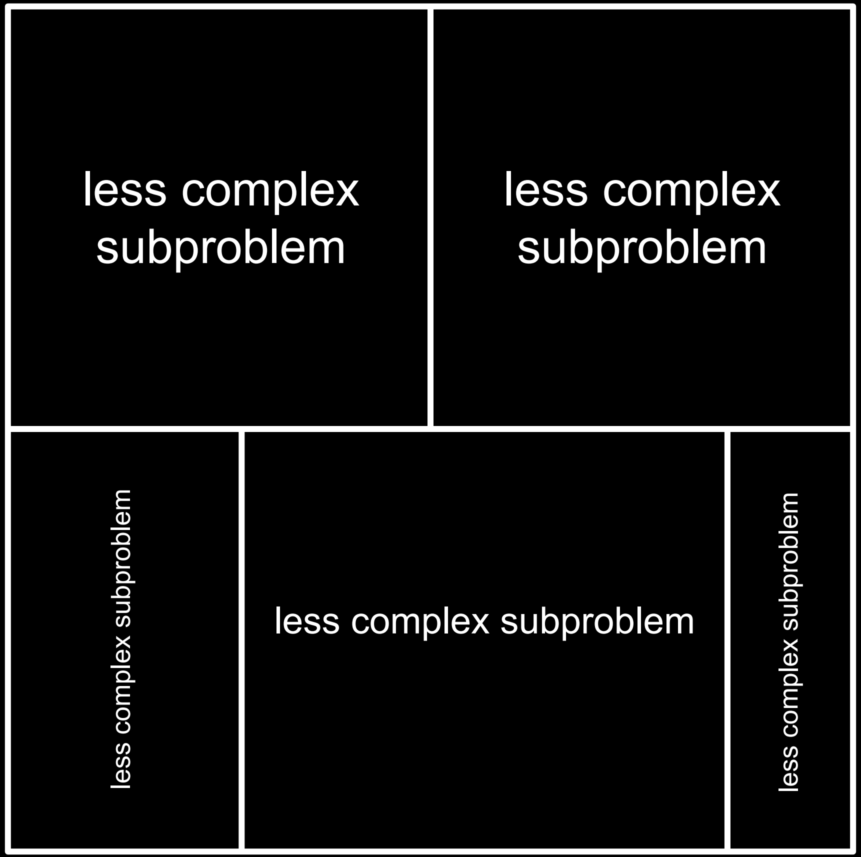

input
ChatGPT
output

problem solving strategies

problem decomposition

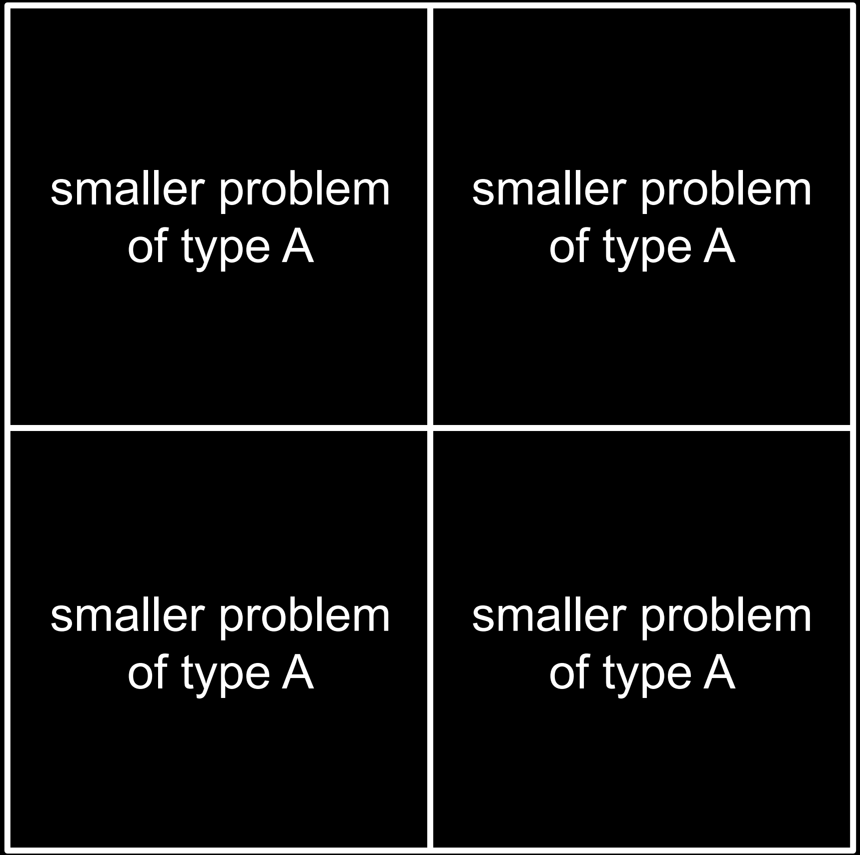
large and complex problem


less complex subproblem
less complex subproblem
less complex subproblem


less complex subproblem
less complex subproblem
less complex subproblem
less complex subproblem
less complex subproblem

divide and conquer

large and complex problem of type A


smaller problem of type A
smaller problem of type A
smaller problem of type A
smaller problem of type A

even smaller problem of type A	even smaller problem of type A
even smaller problem of type A	even smaller problem of type A
even smaller problem of type A	even smaller problem of type A
even smaller problem of type A	even smaller problem of type A


sorted list +
element
search()
yes / no

is 67 a prime number?

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

linear search

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

linear search

~~2~~, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

linear search

~~2~~, ~~3~~, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

linear search

~~2~~, ~~3~~, ~~5~~, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

linear search

~~2~~, ~~3~~, ~~5~~, ~~7~~, ~~11~~, ~~13~~, ~~17~~, ~~19~~, ~~23~~, ~~29~~, ~~31~~, ~~37~~, ~~41~~,
~~43~~, ~~47~~, ~~53~~, ~~59~~, ~~61~~, 67, 71, 73, 79, 83, 89, 97

⬆

19 steps… can't we do better?

~~2~~, ~~3~~, ~~5~~, ~~7~~, ~~11~~, ~~13~~, ~~17~~, ~~19~~, ~~23~~, ~~29~~, ~~31~~, ~~37~~, ~~41~~,
~~43~~, ~~47~~, ~~53~~, ~~59~~, ~~61~~, 67, 71, 73, 79, 83, 89, 97

large and complex problem

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

large and complex problem

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

smaller problem

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41

smaller problem

43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

binary search

```
2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41,
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97
```

binary search

67 != 41

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, ~~41~~,
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

binary search

67 > 41

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, ~~41~~,
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

binary search

67 > 41

⬇

~~2~~, ~~3~~, ~~5~~, ~~7~~, ~~11~~, ~~13~~, ~~17~~, ~~19~~, ~~23~~, ~~29~~, ~~31~~, ~~37~~, ~~41~~,
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

binary search

~~2~~, ~~3~~, ~~5~~, ~~7~~, ~~11~~, ~~13~~, ~~17~~, ~~19~~, ~~23~~, ~~29~~, ~~31~~, ~~37~~, ~~41~~,
43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97

⬆

67 != 71

binary search

~~2~~, ~~3~~, ~~5~~, ~~7~~, ~~11~~, ~~13~~, ~~17~~, ~~19~~, ~~23~~, ~~29~~, ~~31~~, ~~37~~, ~~41~~,
43, 47, 53, 59, 61, 67, ~~71~~, 73, 79, 83, 89, 97

⬆

67 != 71

binary search

~~2~~, ~~3~~, ~~5~~, ~~7~~, ~~11~~, ~~13~~, ~~17~~, ~~19~~, ~~23~~, ~~29~~, ~~31~~, ~~37~~, ~~41~~,
43, 47, 53, 59, 61, 67, ~~71~~, ~~73~~, ~~79~~, ~~83~~, ~~89~~, ~~97~~

⬆

67 < 71

binary search

~~2~~, ~~3~~, ~~5~~, ~~7~~, ~~11~~, ~~13~~, ~~17~~, ~~19~~, ~~23~~, ~~29~~, ~~31~~, ~~37~~, ~~41~~,
43, 47, 53, ~~59~~, 61, 67, ~~71~~, ~~73~~, ~~79~~, ~~83~~, ~~89~~, ~~97~~

⬆

67 != 59

binary search

~~2~~, ~~3~~, ~~5~~, ~~7~~, ~~11~~, ~~13~~, ~~17~~, ~~19~~, ~~23~~, ~~29~~, ~~31~~, ~~37~~, ~~41~~,
~~43~~, ~~47~~, ~~53~~, ~~59~~, 61, 67, ~~71~~, ~~73~~, ~~79~~, ~~83~~, ~~89~~, ~~97~~

67 > 59

binary search

~~2~~, ~~3~~, ~~5~~, ~~7~~, ~~11~~, ~~13~~, ~~17~~, ~~19~~, ~~23~~, ~~29~~, ~~31~~, ~~37~~, ~~41~~,
~~43~~, ~~47~~, ~~53~~, ~~59~~, 61, 67, ~~71~~, ~~73~~, ~~79~~, ~~83~~, ~~89~~, ~~97~~

⬆

67 = 67

binary search

~~2~~, ~~3~~, ~~5~~, ~~7~~, ~~11~~, ~~13~~, ~~17~~, ~~19~~, ~~23~~, ~~29~~, ~~31~~, ~~37~~, ~~41~~,
~~43~~, ~~47~~, ~~53~~, ~~59~~, ~~61,~~ 67, ~~71~~, ~~73~~, ~~79~~, ~~83~~, ~~89~~, ~~97~~

⬆

67 = 67

3 splits → much better

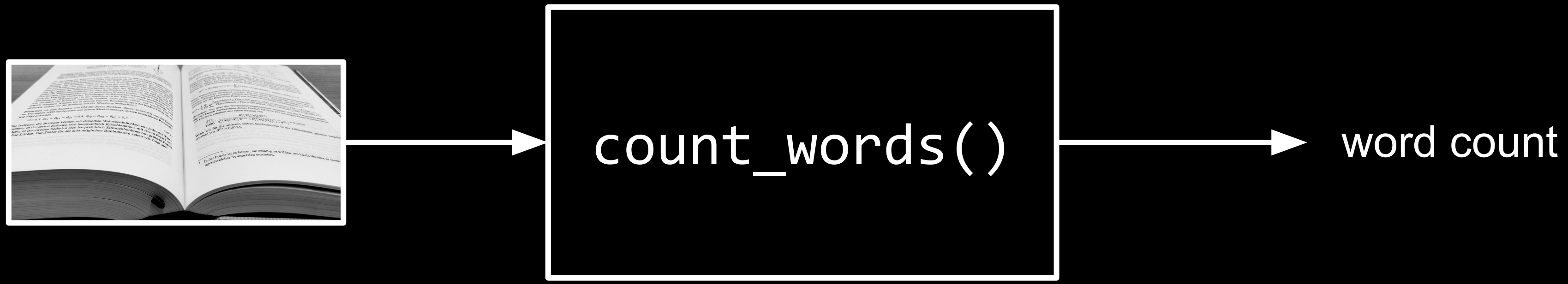
~~2~~, ~~3~~, ~~5~~, ~~7~~, ~~11~~, ~~13~~, ~~17~~, ~~19~~, ~~23~~, ~~29~~, ~~31~~, ~~37~~, ~~41~~,
~~43~~, ~~47~~, ~~53~~, ~~59~~, ~~61,~~ 67, ~~71~~, ~~73~~, ~~79~~, ~~83~~, ~~89~~, ~~97~~

⬆

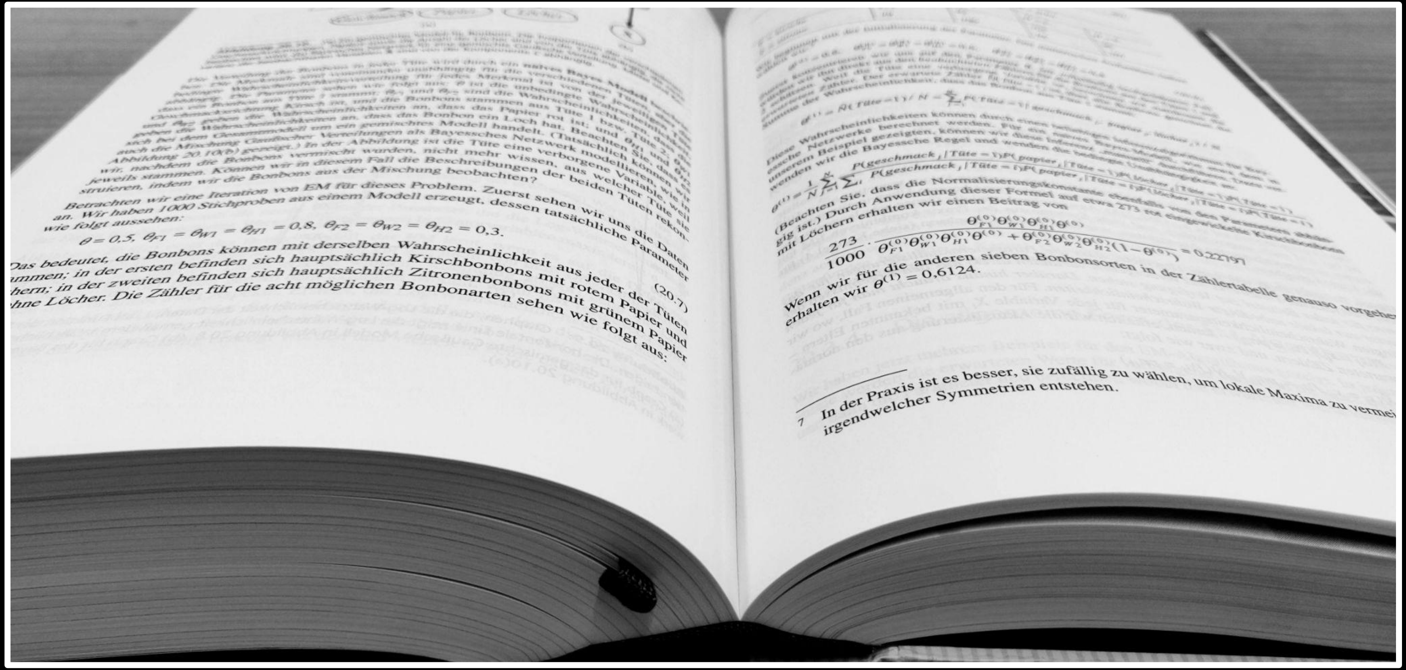
67 = 67

🤔

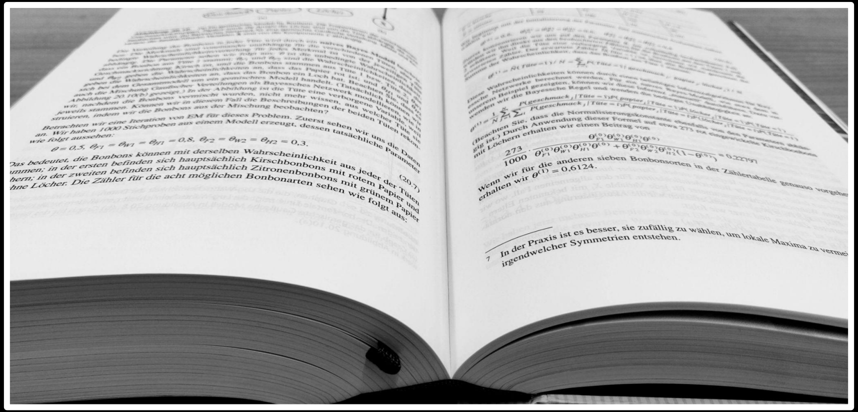
how efficient are linear and binary search in general?


count_words()
word count

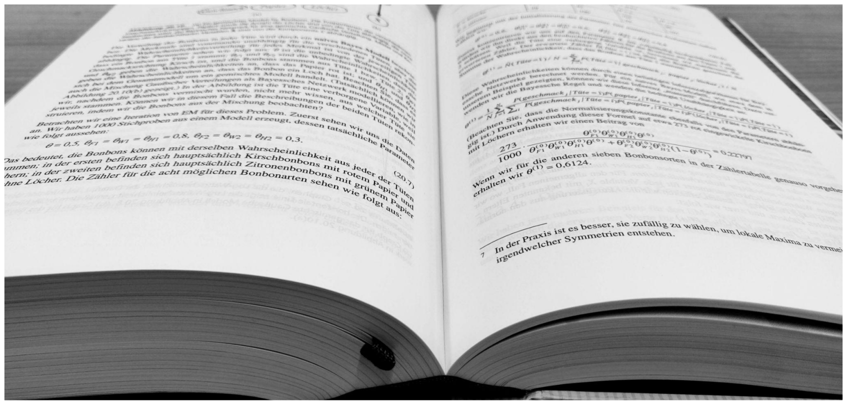
how many words are in the book?



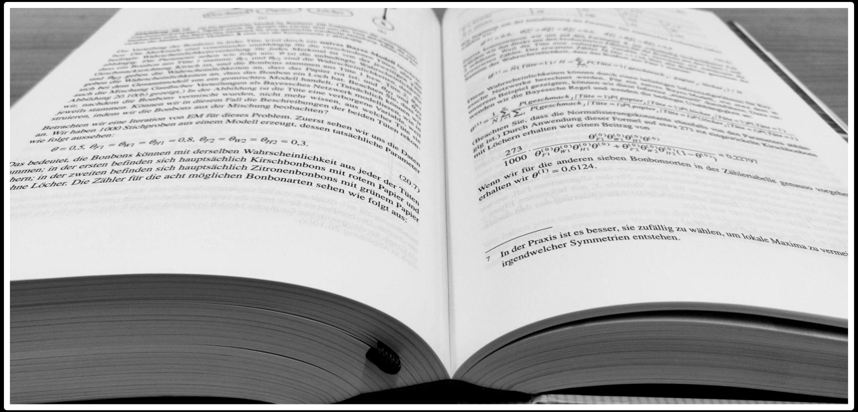
strategies, anyone?



Betrachten wir eine Iteration von EM für dieses Problem. Zuerst sehen wir uns die Daten
an. Wir haben 1000 Stichproben aus einem Modell erzeugt, dessen tatsächliche Parameter
wie folgt aussehen:
$\theta = 0{,}5,\ \theta_{F1} = \theta_{W1} = \theta_{H1} = 0{,}8,\ \theta_{F2} = \theta_{W2} = \theta_{H2} = 0{,}3.$ (20.7)
Das bedeutet, die Bonbons können mit derselben Wahrscheinlichkeit aus jeder der Tüten
in der ersten befinden sich hauptsächlich Kirschbonbons mit rotem Papier und
in der zweiten befinden sich hauptsächlich Zitronenbonbons mit grünem Papier
Löcher. Die Zähler für die acht möglichen Bonbonarten sehen wie folgt aus:
Wenn wir für die anderen sieben Bonbonsorten in der Zählertabelle genauso vorgehe
erhalten wir $\theta^{(1)} = 0{,}6124$.
7 In der Praxis ist es besser, sie zufällig zu wählen, um lokale Maxima zu verme
irgendwelcher Symmetrien entstehen.

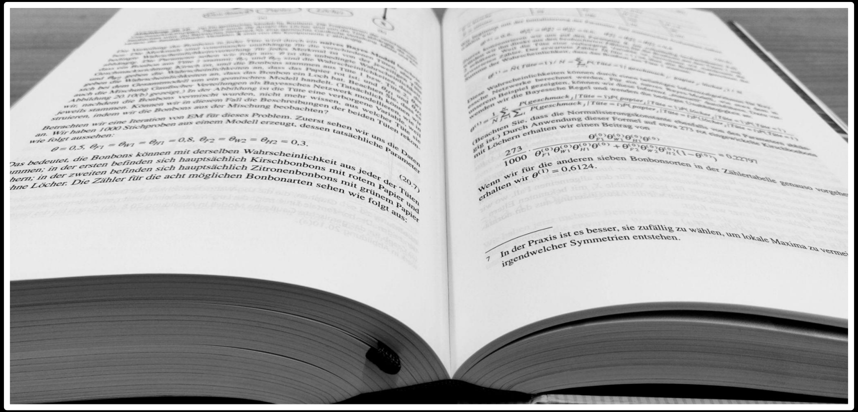

Wenn wir für die anderen sieben Bonbonsorten in der Zählertabelle
In der Praxis ist es besser, sie zufällig zu wählen, um lokale Maxima zu

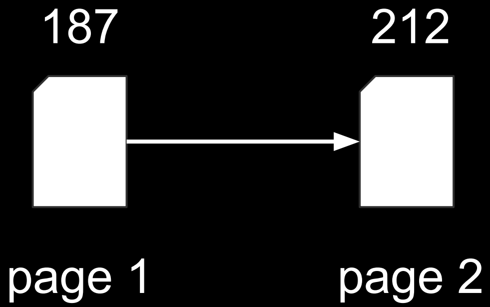
page 1

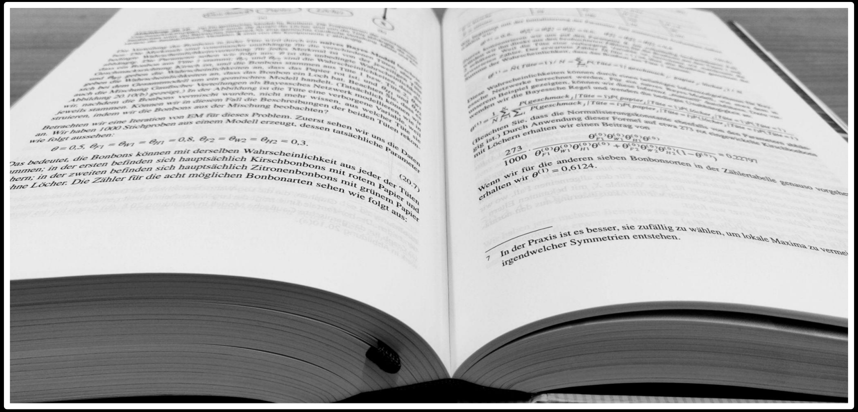

Betrachten wir eine Iteration von EM für dieses Problem. Zuerst sehen wir uns die Daten
an. Wir haben 1000 Stichproben aus einem Modell erzeugt, dessen tatsächliche Parameter
Das bedeutet, die Bonbons können mit derselben Wahrscheinlichkeit aus jeder der Tüten
Wenn wir für die anderen sieben Bonbonsorten in der Zählertabelle genauso
erhalten wir $\theta^{(1)} = 0{,}6124$.
In der Praxis ist es besser, sie zufällig zu wählen, um lokale Maxima zu
irgendwelcher Symmetrien entstehen.

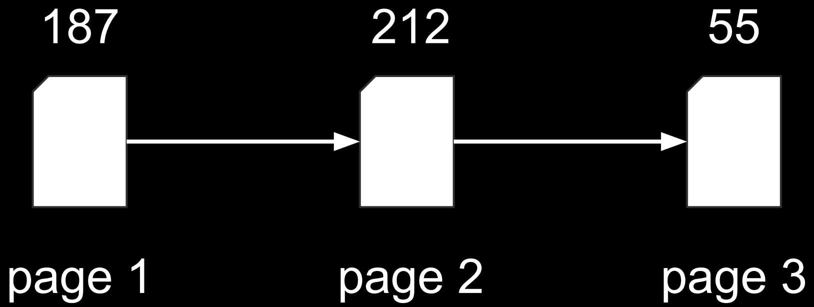
187

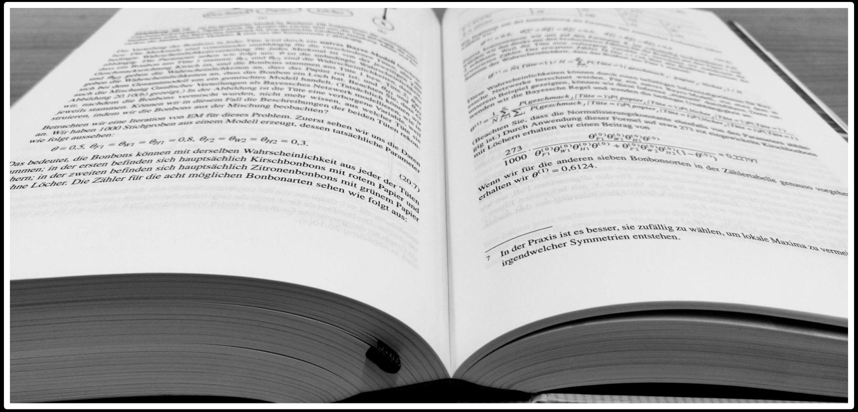
page 1

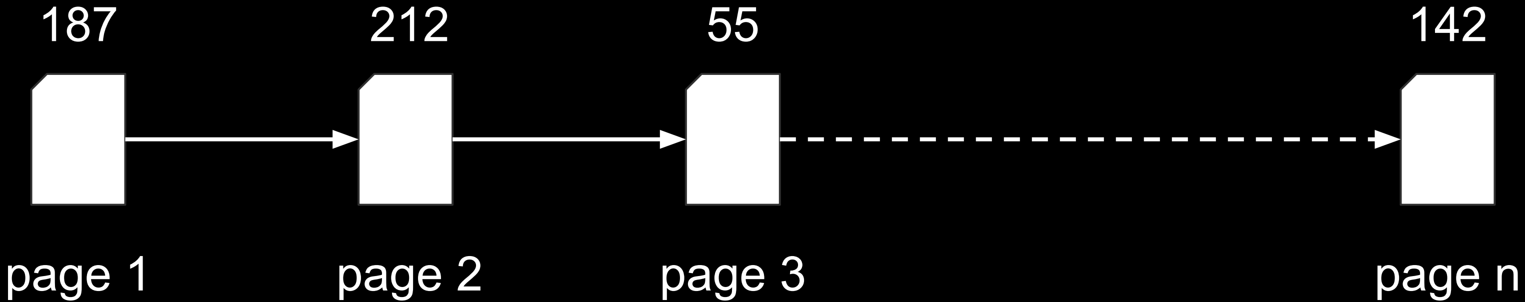

In der Praxis ist es besser, sie zufällig zu wählen, um lokale Maxima zu
irgendwelcher Symmetrien entstehen.

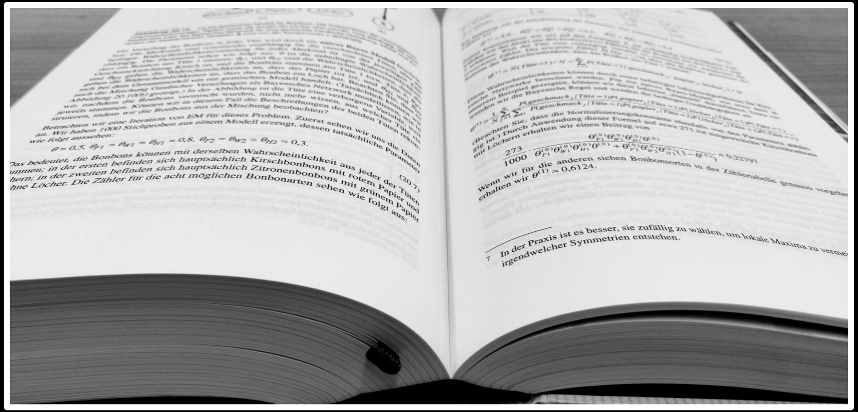

187
212
page 1
page 2




187
212
55
page 1
page 2
page 3




187
212
55
142
page 1
page 2
page 3
page n



n = 1327 pages

Ø 2:23 minutes per page

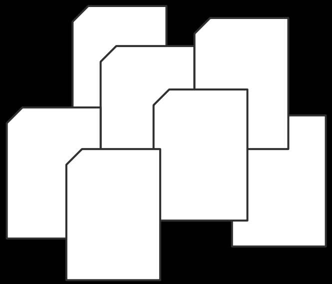
~ 52.34 hours

divide and conquer

+

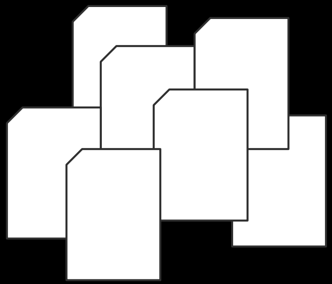
?

pages 1 - 700

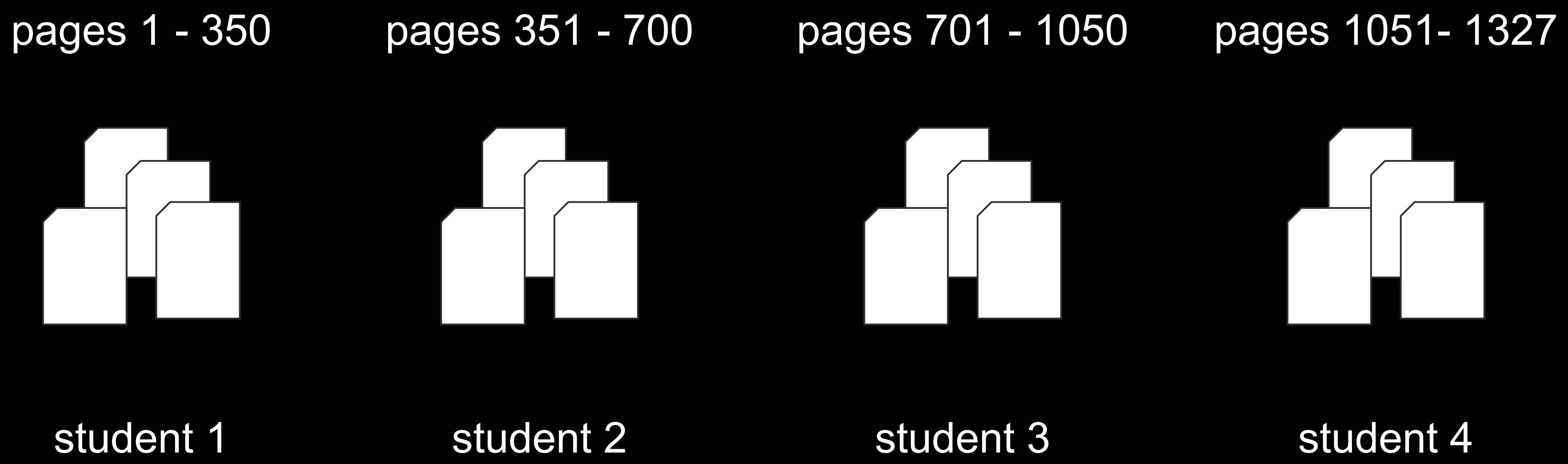


student 1

pages 701 - 1327



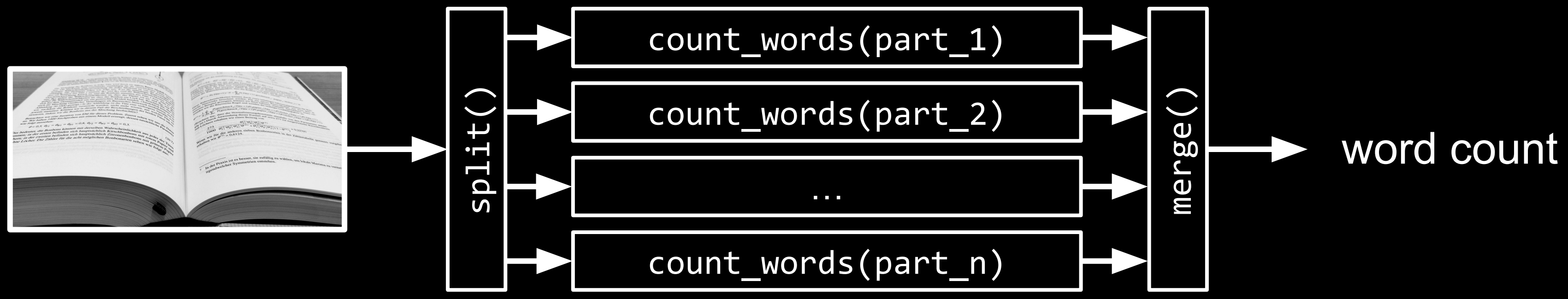
student 2


pages 1 - 350
pages 351 - 700
pages 701 - 1050
pages 1051- 1327
student 1
student 2
student 3
student 4

divide and conquer

+

distribution and parallelization


split()
count_words(part_1)
count_words(part_2)
...
count_words(part_n)
merge()
word count