

Don't need to
do EVERY
Side Quest

Just make sure you
do ALL the main missions

POST ON
FORUM
(Reflections)
+XP


Tutorials Scheduled


Remedial Lessons

Today's Agenda

- Clarifications
- Count Change
 - Recursion
 - Order of Growth
- Higher-order Functions
 - Generalizing Common Patterns
 - Functions as arguments


```
beside (pic1, pic2)
```

```
beside (pic) 
```

```
beside (p1, p2, p3) 
```

Conditional

Form 2

if expr:

 statement(s)

else:

 statement(s)

```
if a > 0
print( 'a > 0!' )
else:
    print( 'a <=
0' )
```

missing colon ☹️

no indentation! ☹️

What is **pass**?

Do nothing 😊

Importing Modules

Remember?

```
from runes import *
```

Insight: Often convenient to have code in different files for code reuse

Importing Modules

- `import X` imports the module `X`
 - Can use `X.name` to refer to objects in `X`
- `from X import *` imports the module `X`, and creates references to all *public* objects in `X`
 - Can use a plain *name*, i.e. `something` to refer to objects defined in module `X`. No need `X.something`
- `from X import a, b, c` imports the module `X`, and creates references to the specified objects.
 - Can now use `a` and `b` and `c` in your program.

Problem

Make change for \$1, using coins
50¢, 20¢, 10¢, 5¢, 1¢

(assume unlimited number of coins
available)

e.g. 50¢ + 50¢,

50¢ + 20¢ + 20¢ + 10¢,

20¢ + 20¢ + 20¢ + 20¢ + 20¢,

etc.

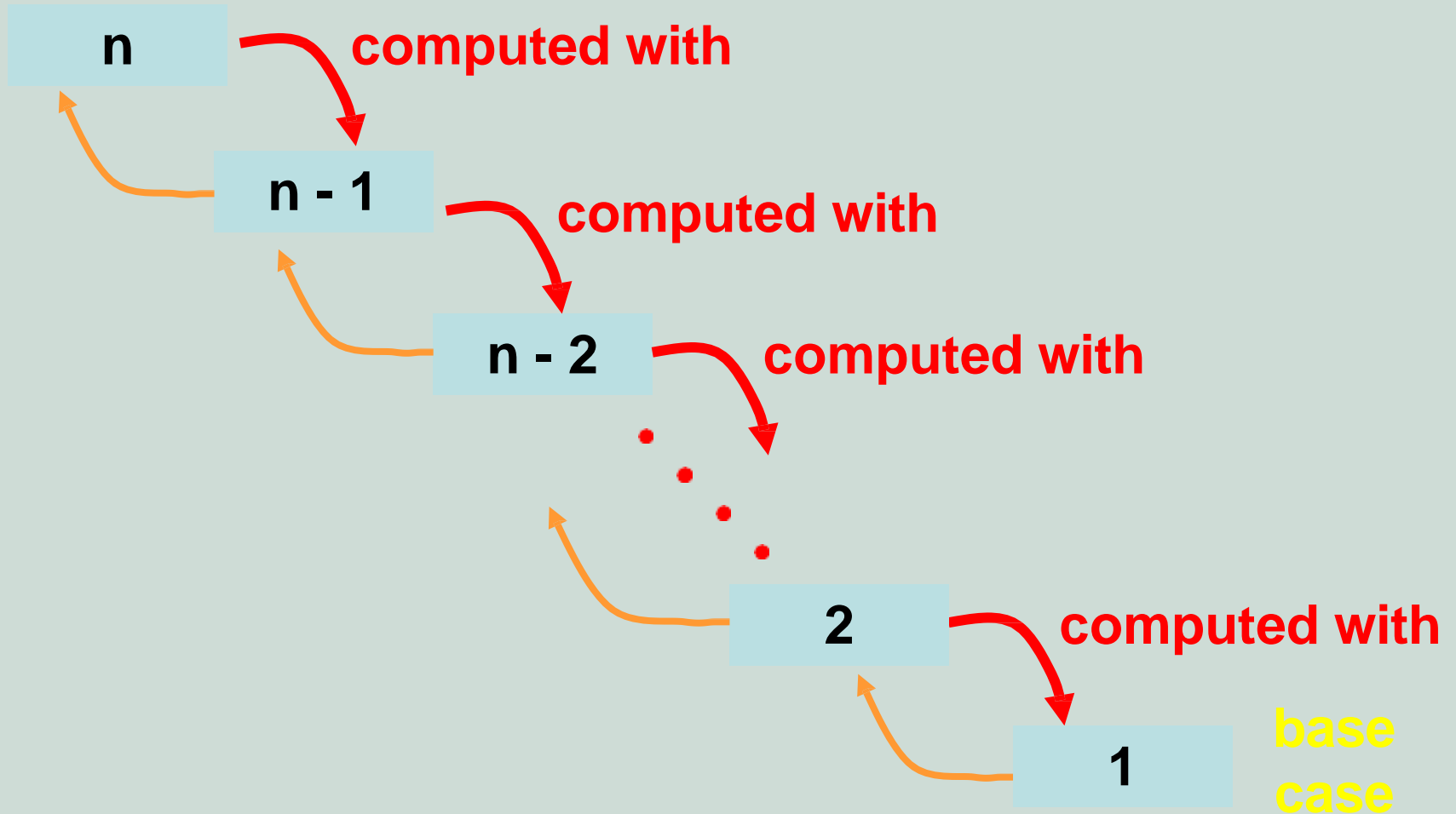
Counting change

How many
ways to do it?

Recap: Recursion

- Express (divide) a problem into one or more smaller similar problems
- Solve the problem for a simple (base) case

Recursion



Base cases

- If $a = 0$, only 1 way to make change.
- If $a < 0$, there is no way to make change, i.e. 0
- If $n = 0$, then no way to make change.

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/295021041301011041>