

Outline for March 7, 2018

Homework: due March 8, 2018 at 11:59pm

1. Modifying parameter lists
 - a. Not directly [*modpar1.py*]
 - b. Using lists [*modpar2.py*]
 - c. Why it works
2. Pattern matching
 - a. Regular expressions
 - b. Atoms: letters, digits
 - c. Match any character except newline: `.`
 - d. Match any of a set of characters: `[0123456789]`, `^[0123456789]`, `[0-9]`
 - e. Repetition: `*`, `?`, `+`, `{m,n}`; greedy matching; put `?` after and they match as few characters as possible
 - f. Match start, end of string: `^`, `$`; `$` matches end of line, also
 - g. Grouping: `(,)`
 - h. Escape metacharacters: `\`
3. Useful functions/methods
 - a. `re.compile(str)` compiles the pattern
 - b. `pc.match(str)` returns None if compiled pattern `pc` does not match beginning of string `str`
 - c. `pc.search(str)` returns None if pattern `pc` does not match any part of string `str`
 - d. `pc.findall(str)` returns a list of substrings of the string `str` that match the pattern `pc`
 - e. `pc.group(str)` returns the substring of the string `str` that the pattern `pc` matches
 - f. `pc.start(str)` returns the starting position of the match
 - g. `pc.end(str)` returns the ending position of the match
 - h. `pc.start(str)` returns the starting position of the match
4. Useful abbreviations
 - a. `\d` matches any digit; same as `[0-9]`
 - b. `\s` matches any space character; same as `[\t\n\r\f\v]`
 - c. `\w` matches any alphanumeric character and underscore; same as `[a-zA-Z0-9_]`
 - d. `\D` matches any character *except* a digit; inverse of `\d`
 - e. `\S` matches any character *except* a space character; inverse of `\s`
 - f. `\W` matches any character *except* an alphanumeric character orr underscore; inverse of `\w`