3.列表

（xc）Python列表

邓 俊 辉

deng@tsinghua.edu.cn

Python List

❖ **在Python中，**<u>List</u>**属于内置的标准数据类型**

❖ box = ['pencil', 'pen', 'ruler', 'rubber']; print box

   # ['pencil', 'pen', 'ruler', 'rubber']

❖ for item in box: print item,

   # pencil pen ruler rubber

❖ box.reverse()

   for item in box: print item,

   # rubber ruler pen pencil

❖ box.sort()

   for item in box: print item,

   # pen pencil rubber ruler

Python List

❖ `for i in range(0, len(box)):` # [0, n)

    `print box[i],`

  `# pen pencil rubber ruler`

❖ `for i in range(len(box)-1, -1, -1):` # [n-1, -1)

    `print box[i],`

  `# ruler rubber pencil pen`

❖ `for i in range(-1, -len(box)-1, -1):` # [-1, -n-1)

    `print box[i],`

  `# ruler rubber pencil pen`

Python List

- bag = ['data structures', 'calculus', box, 2012012012]

  print bag

  # ['data structures', 'calculus',

    ['pen', 'pencil', 'rubber', 'ruler'], 2012012012]

- for item in bag: print item,

  # data structures calculus

    ['pen', 'pencil', 'rubber', 'ruler'] 2012012012

- for item in bag[2]: print item,

  # pen pencil rubber ruler

- for item in bag[2][1:3]: print item,

  # pencil rubber

reverse()

❖ def reverse_1(L): # 循位置访问？

　　for i in range(0, len(L)): # 对[0, n)内的每个i，依次

　　　　L.insert(i, L.pop()) # 将末元素转移至位置i

　　return L # 最终即得倒置后的列表

❖ def reverse_2(L): # 循秩访问？

　　i, j = 0, len(L) - 1 # 从首、末元素开始

　　while i < j: # 依次令对称的L[i]及L[n-1-i]

　　　　L[i], L[j] = L[j], L[i] # 互换，然后

　　　　i, j = i + 1, j - 1 # 考查下一对元素

　　return L # 最终即得倒置后的列表

❖ 哪个版本效率更高？实测结果如何解释？