Buğra Gedik's Courses @ Bilkent University

cs315:recursion_iteration

Converting Recursion into Iteration

Consider traversal:

```
1  def postTraverse(x):
2      for i in xrange(1,len(x)):
3          postTraverse(x[i])
4      print x[0]
5
6  def postTraverse2(x):
7      stack = [(x,0)]
8      while len(stack)>0:
9          (n, i) = stack[-1]
10         if i==len(n)-1:
11             del stack[-1]
12             print n[0]
13         else:
14             stack[-1] = (n, i+1)
15             stack.append((n[1+i],0))
16
17 def postTraverse3(x):
18     stack = [(x,0)]
19     while len(stack)>0:
20         (n, i) = stack[-1]
21         if i==len(n)-1:
22             del stack[-1]
23             yield n[0]
24         else:
25             stack[-1] = (n, i+1)
26             stack.append((n[1+i],0))
27
28 class PostTraverser:
29     def __init__(self, x):
30         self.stack = [(x,0)]
31     def __iter__(self):
32         return self
33     def next(self):
34         while len(self.stack)>0:
35             (n, i) = self.stack[-1]
36             if i==len(n)-1:
37                 del self.stack[-1]
38                 return n[0]
39             else:
40                 self.stack[-1] = (n, i+1)
41                 self.stack.append((n[1+i],0))
```

```
            raise StopIteration

x =
['A',
    ['B',
      ['E'],
      ['F']
    ],
    ['C',
       ['G',
          ['I']
       ],
       ['H',
          ['J'],
          ['K'],
          ['L']
       ]
    ],
    ['D']
]


for c in postTraverse3(x):
    print c

for c in PostTraverser(x):
    print c
```

cs315/recursion_iteration.txt · Last modified: 2013/12/24 17:40 by bgedik