Homework 3 - Q8

Team 2

2019.03.19

Question

> A rod-cutting problem is said that, given a rod and a table of prices. How to cut a steel rod of length n into pieces in order to maximize the revenue r ? Suppose the rod lengths are always an integer. Given a rod of length 10, how to cut it to get the maximum revenue? You need to show your steps and find the maximum revenue.
>
> (Hint: You can use the recursive function in PPT Unit 5.)

Concept

For rod cut method $T(n)$ and revenue function R

- Using divide-and-conquer:

$$T(n) = \{i_n\} \cup T(n - i_n),$$

where

$$i_n = \min_{i \in \mathbb{Z}_n} R(\{n - i\}) + R(T(i)),$$

and treated $T(n)$ as a sequence.

Algorithm (revenue)

```python
def getRevenue(ns):
    revenues = [2, 7, 9, 11, 13, 15, 17, 17, 19, 20]
    r = 0
    for i in ns:
        if i == 0:
            r += 0
        if i > len(revenues) or i < 0:
            raise ValueError("i out of range.")
        else:
            r += revenues[i-1]
    return r
```

Algorithm (divide-and-conquer)

```python
from numpy import argmax

def rodcut(n):
    global cMem # with memoization
    if n == 0: return []
    if cMem[n-1]: return cMem[n-1]
    cuts, revenues = [], []
    for i in range(n):
        c = [n-i] + rodcut(i)
        cuts.append(c)
        revenues.append(getRevenue(c))

    argmaxCuts = cuts[argmax(revenues)]
    cMem[n-1] = argmaxCuts
    return argmaxCuts
```

Algorithm (main function)

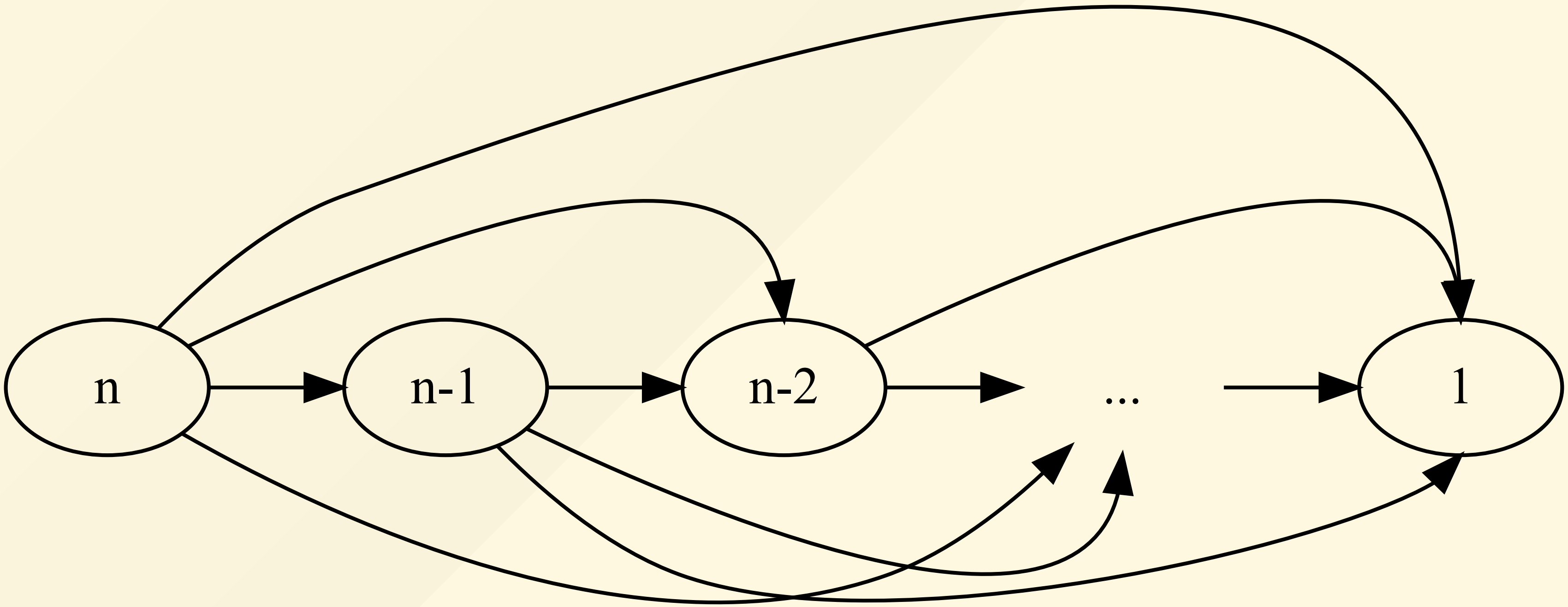
```python
n = 10
cMem = [None for i in range(n)]

argmax = rodcut(n) # best partition
max = getRevenues(argmax)  # best revenue
```

This algorithm is of class $O(n^2)$;

$$(n-1)+(n-2)+\cdots+1=\frac{(n-1)n}{2}=O(n^2).$$

(see the next page for more information)

$$n = 10$$



Thank you for your kind attention

Questions & comments are welcome.