Introduction to algorithms Fall 2017

IFT2125-6001 TA: Maëlle Zimmermann

Demonstration 1

1

Question: Implement in python an algorithm to compute the least common multiple of two numbers a and b.

Solution: The smallest common multiple (ppcm) of two numbers a and b is given by the product a × b divided by the greatest common divisor (pgcd) of a and b. We can implement the following two algorithms in python.

```
#Euclid algorithm for larger common divisor
def gcd (a, b):
    while b:
        a, b = b, a% b
    return a

#Algorithm for smaller common multiple
def lcm (a, b):
    // return a * b gcd (a, b)
```

2

Question: Declare a list, a tuple and a set containing the elements 1,2,3,4 in python and state the main differences. Implement a method of sorting a list.

Solution: A set is an unordered set of elements. A list and a tuple are a sequence of elements, so each element corresponds to an index. It should be noted that python the first element is indexed by 0 and not 1. The main difference between a tuple and a list is that a tuple can only be changed once it is declared.

```
list1 = [1,2,3,4]
```

```
TUP1 = (1,2,3,4)
set1 = {1,2,3,4}

#You can also declare a tuple or set from a list
tup2 = tuple ([1,2,3,4])
set2 = set ([1,2,3,4])

# Insertion sorting algorithm
def InsertionSort (alist)
    for i in range (1, len (alist))
        x = alist [i]
        j = i-1
        while j> = 0 and x <alist [j]
            alist [j + 1] = alist [j]
            j = j-1
        alist [j + 1] = x
    return alist
```

3

Question: Implement an algorithm in python to compute det (A) naively.

Solution: The formula naıve to calculate the determinant of a matrix A of size m × m East:

$$\det(A) = \sum_{j=1}^{m} (-1)^{i+j} a_{ij} \det(A_{i,j})$$

o`ua $_{ij}$ is the j $_{th}$ element of the ˵ i $_{th}$ row of A, and A $_{i,j}$ is the matrix obtained by deleting ˋi th row and j $_{th}$ column of A.ˋ

The line i in the above formula can be chosen randomly. We take defaults to the first line of A, so we set i = 0 in the python algorithm. We 's implement a function for computing the sub-matrix A $_{i,j,}$ and a function that calculates recursively the determinant of matrix A.

```
def submatrix (A, i, j):
    return [[A [x] [y] for y in range (len (A)) if y! = j]
    for x in range (len (A)) if x! = i]

#Analytical algorithm for calculating the determinant
def det (A):
    i = 0 #indice fixed line
    s = 0
    if len (A) == 1:
```

```
    return A [0] [0]
else:
    for j in range (len (A)):
        (A, i, j)) is a subgroup of A (i, j)
        s return
```

4

Question: Prove that:

1. $n^2 + n \in O(n^3)$
2. $n^2 \in \Omega(n \log(n))$
3. $2^{n+1} \in \Theta(2^n)$
4. $n^6 - n^5 + n^4 \in \Theta(n^6)$
5. $\log(n) \in O(\sqrt{\text{not}})$

Solution: There are often several ways to do this. We will see some. Let us first note that we can use the following implications:

i. $\lim_{n \to \infty} f(n) / g(n) \in R_+ \Rightarrow O(f(n)) = O(g(n))$.

ii. $\lim_{n \to \infty} f(n) / g(n) = 0 \Rightarrow O(f(n)) \subset O(g(n))$.

iii. $\lim_{n \to \infty} f(n) / g(n) = +\infty \Rightarrow \Omega(f(n)) \subset \Omega(g(n))$.

1. We can simply use the definition of O (f (n)):

$$O(f(n)) = \{t: N \to R^* \mid \exists n_0 \in N, \exists c \in R^+: \forall n \geq n_0, t(n) \leq cf(n)\}$$

$$\underbrace{n^2 + n \leq n^2 + n^2}_{\forall n \geq 1} \quad \underbrace{2n = 2n^2 \leq ^3}_{\forall n \geq 1} .$$

Thus $\exists n_0 = 1$ and $c = 2$ such that $\forall n \geq n_0$, we have $n^2 + n^3 \leq cn$, and thus ${}_2 n + n \in O(n^3)$.

2. The Hospital's Rule is used:

$$\lim_{n \to \infty} f(n) / g(n) = \lim_{x \to \infty} f(x) / g(x)$$

The limit is calculated:

$$\lim_{n \to \infty} \frac{n^2}{n \log(n)} = \lim_{n \to \infty} \frac{\text{not}}{\log(n)} = \lim_{n \to \infty} \frac{1}{1 / n} = +\infty.$$

So $\Omega(n^2) \subset \Omega(n \log(n))$, so $n^2 \in \Omega(n \log(n))$.

3. The limit is calculated:

$$\lim_{n \to \infty} \frac{2^{n+1}}{2^n} = \lim_{n \to \infty} \frac{2}{1} = 2.$$

Thus $O(2^{n+1}) = O(2^n)$, so $2^{n+1} \in \Theta(2^n)$.

4. Alternatively to the calculation of a limit which would require several rules of the hospital, we can do:

$$O(n^6 - n^5 + n^4) = O\left(\frac{1}{2} n^6 + \left(\frac{1}{2} n^6 - n^5\right) + n^4\right)$$
$$= O\left(\max\left\{\frac{1}{2} n^6, \frac{1}{2} n^6 - n^5, n^4\right\}\right) \quad \text{for } 1/2n^6 - n^5 \geq 0, \forall n \geq 2$$
$$= O\left(\frac{1}{2} n^6\right) = O(n^6).$$

And $n^6 - n^5 + n^4 \in \Theta(n^6)$.

5. Limit is calculated using the Hospital's Rule:

$$\lim_{n \to \infty} \frac{\log(n)}{\sqrt{n}} = \lim_{n \to \infty} \frac{1/n}{\frac{1}{2\sqrt{n}}} = \lim_{n \to \infty} \frac{2}{\sqrt{n}} = 0.$$

Thus $O(\log(n)) \subset O(\sqrt{n})$ and therefore $\log(n) \in O(\sqrt{n})$.

5

Question: Prove by induction that the permutations (12) and (12 ... m) generate S_m, the set of permutations of {1,2, ... m}.

Solution: We prove the proposal in two parts. First, we prove induction m S_m that is generated by the set of transpositions (permutations which exchange two elements and preserve all others).

Base case: m = 2: The set S_2 contains only the identity permutation and permu- (12). Since Id = (12) (12), this is true.

Induction Step: Let m> 2. Suppose that the proposition is true for S_{m-1}. Is a permutation $\sigma \in S_m$.

- If σ leaves m fixed, then the restriction of σ to {12 ... m - 1} is generated by transpositions (which leave m fixed) by induction hypothesis.
- Otherwise, σ (m) = y = m. Let the transposition τ = (my), then the permutation στ fixed m and as previously it is generated by transpositions. As $\sigma = \sigma\tau\tau^{-1} = (\sigma\tau)\tau$, it is concluded that σ is generated by transposition.

In a second step, we prove that all the transpositions of {1,2, ... m} are generated by (12) and (12 ... m). Indeed, we note first of all that any trans- (kk + 1) where 1 ≤ k ≤ m - 1 is generated by (12) and γ = (12 ... m):

$$\begin{cases} \gamma^{-1}(12)\gamma = (23) \\ \gamma^{-1}(23)\gamma = (34) \\ \dots \\ \gamma^{-1}(m-2\ m-1) = \gamma\,(m-1\ m). \end{cases}$$

Then we prove that any transposition of the form (1 k) for $2 \le k \le m$ is written as a product of transpositions of the preceding type. Indeed, for $k \ge 3$ we have

$$(1\ k) = (k\,k-1)\,(1\ k-1)\,(k\,k-1).$$

Finally, it remains to be noted that any transposition (xy) can be written as product (1 x) (1 y) (1 x) to conclude that any transposition of {1,2, ... m} can be generated by (12) and (12 ... m).