BLG372E Analysis of Algorithms

Homework 2

Ozan Arkan CAN
040090573

PROBLEM 1

In genereal, order of functions like that:

$log(n) \leq n^x \leq r^n \leq n^n$, $\forall x > 0$ and $r > 1$

We can order given functions by this order information.

$F_9 = 2/n < F_{12} = 37 < F_2 = \sqrt{n} < F_1 = n < F_6 = nlog(logn) < F_5 = nlogn < F_8 = nlog(n^2) < F_7 = nlog^2n < F_3 = n^{1.5} < F_4 = n^2 < F_{13} = n^2logn < F_{14} = n^3 < F_{11} = 2^{n/2} < F_{10} = 2^n$

F_9 is a descending function, so its complexity minimum. F_{12} is a constant function and it never grows. Its obvious that relation between $\sqrt{n}$, n , $nlog(logn)$ and $nlogn$. If we apart function F_8 and F_7 like n times $log(n^2)$, log^2n; those part of functions act logaritmic and pow of number. F_3, F_4 , F_{13} and F_{14} functions are ordered by their power and $log(n) < n$ information. F_{11} and F_{10} functions are ordered by their power.

PROBLEM 2

First of all, given algorithm does not work all possible situation. For n = 16 and maximum independent set size = 11, k gets following values:

8, 12, 6, 9, 13, 7, 10, 15, 8

There is infinite loop and exceeding loop counter (log_2n).

Algorithm checks there is an independent set of size k in a graph in the loop. But k is not constant, so for this line complexity is not $O(n^k)$. k can be $O(n)$. Thus, complexity goes $O(n^n)$. I assume algorithm terminates $O(log_2n)$ by ignoring bug of algorithm. In this situation, complexty is $O(log_2n * n^n)$. $(n^2) * 2^n < log_2n * n^n$. Given algorithm is slower than the slides in worst case.

PROBLEM 3

If we investigate each iteration of outer loop, we can see the second loop after outer loop iterates $n - 1$. If we look inner loop iteration by iteration, total number of iteration is 1 for S_1 , 2 for S_2 , 3 for S_3 ,..., n for S_n . Here, S_i belongs to outer loop. Thus, total number of iteration of whole algorithm is;

$(n-1) * 1 + (n-1) * 2 + ... + (n-1) * n$
$(n-1) * \frac{n*(n+1)}{2}$

So complexity is;

$(n-1) * \frac{n*(n+1)}{2} = O(n^3)$

PROBLEM 4

- a) It is true.
  $c_1 * h \leq f \leq c_2 * h \ n \geq n_0$
  $d_1 * h \leq g \leq d_2 * h \ n \geq n_1$
  $c_1 * d_1 * h * h \leq f * g \leq c_2 * d_2 * h * h \ , \ n \geq max(n_0, n_1)$
  $f * g = \theta(h * h)$
  $c_1 * d_1$ and $c_2 * d_2$ are constant.

- b)
  $log_2(n) \leq c * n^{0.5} \ , \ n \geq n_0$
  $n \leq 2^{c*n^{0.5}} \ , \ n \geq n_0$
  $c = 1 \ , \ n_0 = 0 \Rightarrow n \leq 2^{c*n^{0.5}} \ , \ \forall n \ n \geq n_0$