Instruction to candidates

(time allowed 2 hours)

- candidates will be assessed on their best four answers
- if you attempt to answer more than the required number of questions, the marks awarded for the excess questions will be discarded (starting with your lowest mark)
- all logarithms are to the base 2

Question 1

1.A What is a closed form of:

1. **[5 marks]**

$$\sum_{i=0}^{n} 2^i \ ?$$

2. **[10 marks]**

$$\sum_{i=0}^{n} \binom{n}{i} 2^i \ ?$$

Justify both of your answers.

1.B Prove that

$$\binom{n}{k} \geq \binom{n-1}{k},$$

for any $n \geq 1$ and $0 \geq k \geq n$. For what value of k does the equality $\binom{n}{k} = \binom{n-1}{k}$ hold.

[10 marks]

Question 2

2.A Prove, using mathematical induction, that $T(n) = 3^n - 1$, where value $T(n)$ is defined by the recurrence:

$$T(n) = \begin{cases} 2 & \text{for } \; n = 1, \\ 3T(n-1) + 2 & \text{for } \; n > 1, \end{cases}$$

[15 marks]

2.B Is the function $f(x) = x+1$ *bijective* when the domain and the codomain are $\mathbf{N}$? Is it bijective when the domain and the codomain are $\mathbf{Z}$?

Justify your answers. **[10 marks]**

Question 3

3.A Design and write pseudocode finding a *maximum* and a second *maximum* element in array $A[1, \ldots, n]$ of real numbers. The pseudocode should report both extremal values. What is the complexity of your solution? **[15 marks]**

3.B In each case, which function is larger for almost all n (i.e. which function has higher order):

(i)

$$n^{\frac{2000}{2001}} \qquad \frac{n}{\sqrt{\log(n)}}$$

(ii)

$$\log(n^{2001}) \qquad n^{\frac{1}{2001}}$$

(iii)

$$n^{\log(n)} \qquad 2^{2001\log(n)}$$

[6 marks]

3.C What is the value of:

$$\log^*(128)$$

[2 marks]

3.D Which exact complexity is better for realistic values of parameter n,

$$2001 \quad \textit{or} \quad \log(n)?$$

Justify briefly your answer. **[2 marks]**

Question 4

4.A Prove that

$$3n^3 + 4n = \Theta(n^3).$$

[15 marks]

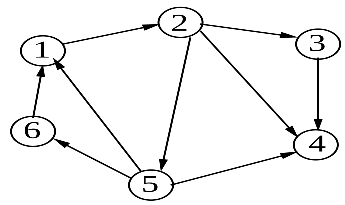
4.B List all 3-combinations of set $\mathcal{S} = \{A, B, C, D, E\}$. Is the number of all 2-combinations of $\mathcal{S}$ larger then the number of all 3-combinations?

[4 marks]

4.C List three different sorting algorithms and comment on their worst case complexities. **[6 marks]**

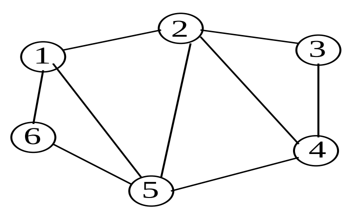
Question 5

5.A Consider the following directed graph G_1:



1. Explain why graph G_1 isn't strongly connected. **[2 marks]**
2. Propose one edge to make G_1 strongly connected. **[2 marks]**
3. What is the in-degree and what is the out-degree of node 2? **[2 marks]**
4. What is the shortest directed path from node 1 to 6? **[2 marks]**
5. What is the longest simple path in G_1? **[2 marks]**

5.B Consider the following (undirected) graph G_2:



1. Draw any spanning tree in G_2. **[5 marks]**
2. How many edges are missing to make G_2 complete? **[2 marks]**
3. What is a diameter of G_2? **[2 marks]**
4. What is the minimum and the maximum degree of G_2? **[2 marks]**
5. What is the longest simple path between nodes 1 and 6? **[2 marks]**
6. Is G_2 planar? **[2 marks]**

Answers to question 1

1.A (1) We use the fact that $\sum_{i=0}^{n} x^i = \frac{x^{n+1}-1}{x-1}$. When we plug in $x = 2$ we get $\frac{2^{n+1}-1}{2-1} = 2^{n+1} - 1$. Thus $\sum_{i=0}^{n} 2^i = 2^{n+1} - 1$.

(2) We use the fact that $(x + y)^n = \sum_{i=0}^{n} \binom{n}{i} x^i \cdot y^{n-i}$. When we plug in $x = 2$ and $y = 1$ we get $(2+1)^n = \sum_{i=0}^{n} \binom{n}{i} 2^i \cdot 1^{n-i} = \sum_{i=0}^{n} \binom{n}{i} 2^i$. Thus $\sum_{i=0}^{n} \binom{n}{i} 2^i = 3^n$.

1.B We know (e.g. from the definition of Pascal's Triangle) that $\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}$. Since all binomial coefficients are non-negative so $\binom{n-1}{k-1}$ is non-negative too. This implies that $\binom{n}{k} \geq \binom{n-1}{k}$.
It remains to check when $\binom{n}{k} = \binom{n-1}{k}$. We use the fact that $\binom{n}{k} = \frac{n!}{k! \cdot (n-k)!}$. And indeed we start with

$$\frac{n!}{k! \cdot (n-k)!} = \frac{(n-1)!}{k! \cdot (n-k-1)!},$$

we divide both sides by $\frac{(n-1)!}{k! \cdot (n-k-1)!}$ and we get

$$\frac{n}{n-k} = 1.$$

This is possible only when the parameter $k = 0$.

Answers to question 2

2.A The prove that property (*) $T(n) = 3^n - 1$ holds is performed in two steps.

1. (Basis) Initially we prove that property (*) holds for small integer $n_0 = 1$. And indeed $T(1) = 2$ from the definition of T, since $3^{n_0} - 1 = 3 - 1 = 2$.

2. (Inductive step) In what follows we assume that property (*) holds for all integers from range $n_0, \ldots, n-1$. We prove now, using the definition of T and inductive assumption that (*) holds for value $n-1$, that property (*) holds also for value n. And indeed

$$T(n) = 3T(n-1) + 2 = 3 \cdot (3^{n-1} - 1) + 2 = 3^n - 3 + 2 = 3^n - 1.$$

Using induction we proved that property (*) holds for all integers.

2.B Function f is *bijective* if it is both *injective* (distinct arguments to f produce distinct values) and *surjective* (range of f is equal to its codomain). It is relatively easy to prove that function f is *injective*. And indeed for any two distinct arguments $x \neq y$ the following holds $f(x) = x+1 \neq y+1 = f(y)$. It remains to check whether function f is *surjective* in any of the two cases.

Case 1 Domain and codomain are $\mathbf{N} = \{0, 1, 2, 3, ..\}$. In this case the range of f is equal to $\{1, 2, 3, ...\}$ since there isn't any natural number x, s.t. $f(x) = x + 1 = 0$. Thus function f isn't surjective as well as bijective when its domain and codomain are $\mathbf{N}$.

Case 2 Domain and codomain are $\mathbf{Z} = \{.., -2, -1, 0, 1, 2, 3, ..\}$. In this case the range of f is equal to its codomain $\mathbf{Z}$ since for any $x \in \mathbf{Z}$, there exist $y = x - 1$, s.t. $f(y) = x$. Thus function f is surjective as well as bijective when its domain and codomain are $\mathbf{Z}$.

Answers to question 3

3.A

	cost	#times
...		
$max1 \leftarrow MAX(A[1], A[2]);\ max2 \leftarrow MIN(A[1], A[2]);$	c_1	1
for $i \leftarrow 3$ **to** n **do**	c_2	$n-1$
if $A[i] > max1$	c_3	$n-2$
then $max2 \leftarrow max1;\ max1 \leftarrow A[i];$	c_4	$\leq n-1$
else if $A[i] > max2$ **then** $max2 \leftarrow A[i];$	c_5	$\leq n-1$
▷ $max1$ contains a maximum value and		
▷ max contains a second maximum value.		
...		

The total cost of the solution is bounded by:

$$(c_2 + c_3 + c_4 + c_5)n + c_1 - c_3 - c_4 - c_5 = O(n).$$

3.B (i) $n^{\frac{2000}{2001}} < \frac{n}{\sqrt{\log n}}$, (ii) $\log(n^{2001}) < n^{\frac{1}{2001}}$, (iii) $n^{\log n} > 2^{2001 \log n}$.

3.C $\log^*(128) = 4$, since $2^{2^2} = 16$ and $2^{2^{2^2}} > 128$.

3.D $2001 > \log(n)$ for all realistic values of n, since inequality holds for all $n < 2^{2001}$ which is much larger then the number of all atoms in the solar system.

Answers to question 4

4.A We start with prove that $3n^3 + 4n = O(n^3)$. We have to show that there is a constant $c > 0$ and integer n_0, s.t. $0 \leq 3n^3 + 4n \leq cn^3$, for all integer $n \geq n_0$. The inequality holds when $n^2(c-3) \geq 4$ and equivalently $n \geq \sqrt{\frac{4}{c-3}}$. If we take $c = 4$ the inequality holds for all $n \geq 2$, thus we can take $n_0 = 2$.

Later we prove that $3n^3 + 4n = \Omega(n^2)$. We have to show that there is a constant $c > 0$ and integer n_0, s.t. $0 \leq cn^3 \leq 3n^2 + 4n$, for all integer $n \geq n_0$. And indeed if we take $c = 1$ we get $n^3 \leq 3n^3 + 4n$, which is true for any integer $n \geq 1$. Thus in this case $c = 1$ and $n_0 = 1$.

Finally since $3n^3 + 4n = O(n^3)$ and $3n^3 + 4n = \Omega(n^3)$ we get $3n^3 + 4n = \Theta(n^3)$.

4.B The 3-combination (listed in lexicographical order) of set $\mathcal{S} = \{A, B, C, D, E\}$ are: $\{A, B, C\}, \{A, B, D\}, \{A, B, E\}, \{A, C, D\}, \{A, C, E\}, \{A, D, E\}, \{B, C, D\}$, $\{B, C, E\}, \{B, D, E\}$, and $\{C, D, E\}$. There are exactly the same number of 2-combinations and 3-combinations in this case due to the symmetry of a binomial coefficient, i.e. $\binom{n}{k} = \binom{n}{n-k}$, where $n = 5$ and $k = 3$. .

4.C Three sorting algorithms:

1. *Bubble-sort*, its worst case complexity is $O(n^2)$, since it can remove only one *inversion* at a time,

2. *Insertion-sort*, its worst time complexity is $O(n^2)$, it happens when the input is sorted in reverse order,

3. *Merge-sort*, its complexity is $\Theta(n \log n)$, and it doesn't depend on the input data.

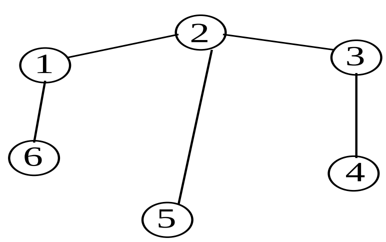
Answers to question 5

5.A

1. Out-degree of node 4 is 0,
2. any out-going edge from node 6, e.g. $4 \to 5$,
3. in-degree is 2, and out-degree is 1,
4. the shortest path is of length 3: $1 \to 2 \to 5 \to 6$
5. longest directed simple path has length 5, $5 \to 6 \to 1 \to 2 \to 3 \to 4$.

5.B

1. Example of a spanning tree:



2. $\frac{6 \cdot 5}{2} - 9 = 6$, missing edges: $1-3, 1-4, 2-6, 3-5, 3-6, 4-6$,
3. the diameter of G_2 is 3 (use a route along a perimeter).
4. the minimum degree is 2 (nodes 3 and 6) and the maximum degree is 4 (nodes 2 and 5),
5. the longest simple path between nodes 1 and 6 is of length 5, see $1-5-2-4-3-6$, and it cannot be longer since we would have a cycle.
6. G_2 is planar since it can be placed on a plane, such that its edges do not cross each other.