Izbirni izpit za 2. stopnjo IŠRM

Primer izpita

1. naloga

Naj bo (x_i) neskončno zaporedje realnih števil, za katero vrsta $\sum_{i=0}^{\infty} x_i$ konvergira. Katera, če sploh katera, od trditev (A)–(C) spodaj **ne sledi**?

A) Zaporedje $(x_i)_i$ konvergira proti 0.

B) Zaporedje absolutnih vrednosti $(|x_i|)_i$ konvergira.

C) Vrsta $\sum_{i=0}^{\infty} x_{2i}$ konvergira.

D) Vse trditve (A)–(C) veljajo.

Suppose $(x_i)_i$ is an infinite sequence of real numbers such that the series $\sum_{i=0}^{\infty} x_i$ converges. Which, if any, of the statements (A)–(C) below **does not follow**?

A) The sequence $(x_i)_i$ converges to 0.

B) The sequence $(|x_i|)_i$ of absolute values converges.

C) The series $\sum_{i=0}^{\infty} x_{2i}$ converges.

D) Statements (A)–(C) all follow.

2. naloga

Za parametra $\alpha, \beta \in \mathbb{R}$ naj bodo podane tri ravnine v trirazsežnem prostoru:

$$\begin{aligned} x + y + z &= 3 \\ x - y &= 1 \\ 2x + (1-\alpha)y + \alpha z &= \beta, \end{aligned}$$

Katera izmed spodnjih trditev **je resnična**?

A) Za $\alpha = 0$ in $\beta = 2$ imajo vse tri ravnine skupno premico.

B) Za $\alpha = 0$ in $\beta = 4$ imajo vse tri ravnine skupno premico.

C) Za $\alpha = 1$ in $\beta = 2$ imajo vse tri ravnine skupno premico.

D) Za $\alpha = 1$ in $\beta = 4$ imajo vse tri ravnine skupno premico.

Consider the following three planes in 3-dimensional space

$$\begin{aligned} x + y + z &= 3 \\ x - y &= 1 \\ 2x + (1-\alpha)y + \alpha z &= \beta, \end{aligned}$$

where $\alpha, \beta \in \mathbb{R}$ are parameters. Which of the following statements **is correct**?

A) For $\alpha = 0$ and $\beta = 2$, the three planes have a common line.

B) For $\alpha = 0$ and $\beta = 4$, the three planes have a common line.

C) For $\alpha = 1$ and $\beta = 2$, the three planes have a common line.

D) For $\alpha = 1$ and $\beta = 4$, the three planes have a common line.

3. naloga

Dana je diferencialna enačba

$$y''(x) + \alpha y'(x) + 2y(x) = f(x), \quad y(0) = 1, \quad y'(0) = 0,$$

kjer je $\alpha \in \mathbb{R}$ in $f : \mathbb{R} \to \mathbb{R}$. Katera od naslednjih trditev **je napačna**?

A) Za $\alpha = 0$ in $f(x) = 2$ je rešitev diferencialne enačbe enaka $y(x) = 1$.

B) Za $\alpha = 0$ in $f(x) = 0$ je rešitev diferencialne enačbe enaka $y(x) = \cos(\sqrt{2}x)$.

C) Za $\alpha = -3$ in $f(x) = 0$ je rešitev diferencialne enačbe enaka $y(x) = 2e^{2x}$.

D) Za $\alpha = -3$ in $f(x) = 6e^{-x}$ je rešitev diferencialne enačbe enaka

$$y(x) = e^{-x} - e^{x} + e^{2x}.$$

Consider the differential equation

$$y''(x) + \alpha y'(x) + 2y(x) = f(x), \quad y(0) = 1, \quad y'(0) = 0,$$

where $\alpha \in \mathbb{R}$, and $f : \mathbb{R} \to \mathbb{R}$. Which of the following statements **is not true**?

A) For $\alpha = 0$ and $f(x) = 2$, the solution of the differential equation is $y(x) = 1$.

B) For $\alpha = 0$ and $f(x) = 0$, the solution of the differential equation is $y(x) = \cos(\sqrt{2}x)$.

C) For $\alpha = -3$ and $f(x) = 0$, the solution of the differential equation is $y(x) = 2e^{2x}$.

D) For $\alpha = -3$ and $f(x) = 6e^{-x}$, the solution of the differential equation is

$$y(x) = e^{-x} - e^{x} + e^{2x}.$$

4. naloga

Spomnimo se, da je dvočlena relacija R na množici X *irefleksivna*, kadar velja $\forall x \in X.\neg(x R x)$, in *antisimetrična*, kadar velja $\forall x, y \in X. x R y \wedge y R x \Rightarrow x = y$. Naj bo R poljubna tranzitivna dvočlena relacija na množici X in predpostavimo, da obstajata $x, y \in X$, da velja $x R y$. Kakšna **ne more hkrati** biti relacija R?

A) Refleksivna in simetrična.

B) Refleksivna in antisimetrična.

C) Irefleksivna in simetrična.

D) Irefleksivna in antisimetrična.

Recall that a binary relation R on a set X is said to be *irreflexive* when $\forall x \in X.\neg(x R x)$, and *antisymmetric* when $\forall x, y \in X. x R y \wedge y R x \Rightarrow x = y$. Now let R be an arbitrary transitive binary relation on a set X, and suppose that there exist $x, y \in X$ such that $x R y$. Which of the following is **an impossible combination** of properties for R?

A) Reflexive and symmetric.

B) Reflexive and antisymmetric.

C) Irreflexive and symmetric.

D) Irreflexive and antisymmetric.

5. naloga

Metodo

```
public static int f(int a, int b) {
    ...
}
```

bi radi dopolnili tako, da bo za vse pare celih števil a in b z intervala $[0, 10^3]$ vrnila pravilen rezultat izraza ab. Na katerega od sledečih načinov tega **ne moremo** storiti?

A) `return (b == 0) ? (0) : (f(a + 1, b - 1) + a - b + 1);`

B) `return (b == 0) ? (0) : (3 * f(a, b / 3) + a * b / 3);`

C) `return (b == 0) ? (0) : ((f(a, b >> 1) << 1) + a * (b % 2));`

D) `return (a == 0) ? (0) : (b + f(a - 1, b));`

We would like to complete the method

```
public static int f(int a, int b) {
    ...
}
```

so that it will return the correct result of the expression ab for all pairs of integers a and b from the interval $[0, 10^3]$. Which of the following ways **does not** achieve this goal?

A) `return (b == 0) ? (0) : (f(a + 1, b - 1) + a - b + 1);`

B) `return (b == 0) ? (0) : (3 * f(a, b / 3) + a * b / 3);`

C) `return (b == 0) ? (0) : ((f(a, b >> 1) << 1) + a * (b % 2));`

D) `return (a == 0) ? (0) : (b + f(a - 1, b));`

6. naloga

Podana sta razred `Zival` in njegov podrazred `Pes`. V obeh je definiran konstruktor brez parametrov. **Kaj se zgodi**, če poskušamo sledeči program prevesti in pognati?

```
public class Test {
    public static void main(String[] args) {
        Zival a = new Zival();
        Zival b = new Pes();
        Pes c = new Pes();
        Pes d = (Pes) b;
        Zival e = (Pes) (Zival) c;
        Pes f = (Pes) (Zival) d;
        Zival[] zivali = {a, b, c, d};
    }
}
```

A) Program se ne prevede.

B) Program se prevede, pri izvajanju pa se sproži izjema tipa `ClassCastException`.

C) Program se prevede, pri izvajanju pa se sproži izjema tipa `ArrayStoreException`.

D) Program se prevede in izvede.

You are given a class `Animal` and its subclass `Dog`. Both contain the definition of a constructor without parameters. **What happens** if we try to compile and run the following program?

```
public class Test {
    public static void main(String[] args) {
        Animal a = new Animal();
        Animal b = new Dog();
        Dog c = new Dog();
        Dog d = (Dog) b;
        Animal e = (Dog) (Animal) c;
        Dog f = (Dog) (Animal) d;
        Animal[] animals = {a, b, c, d};
    }
}
```

A) The program does not compile.

B) The program compiles, but an exception of the type `ClassCastException` is thrown when it is run.

C) The program compiles, but an exception of the type `ArrayStoreException` is thrown when it is run.

D) The program compiles and runs.

7. naloga

Butalci so čudni ljudje. Včasih so povsem normalni, včasih se pa skregajo s pametjo prav hudo. Tako so pričeli po cestah Butal zaračunavati mitnino. A bi ne bili Butalci, če ne bi na nekaterih cestah zaračunavali negativno vrednost mitnine — z drugimi besedami, tistemu, ki se je peljal po takšni cesti, so celo plačali. Da bi vse skupaj bilo preglednejše, so pripravili razpredelnico, v katero so vpisali višino mitnine med posameznimi kraji Butal:

	A	B	C	$\boxed{\check{Z}}$	D	E	F	G	H	I
A	0	−94	12	19	72	81	∞	∞	7	−64
B	∞	0	83	∞	∞	55	∞	46	9	∞
C	−12	−57	0	37	36	28	86	95	36	97
$\boxed{\check{Z}}$	−86	−50	96	0	79	∞	−86	∞	∞	∞
D	−21	∞	∞	∞	0	∞	∞	41	67	54
E	−61	−1	∞	65	∞	0	41	∞	53	∞
F	∞	78	∞	71	28	88	0	6	−47	100
G	∞	−45	∞	16	53	∞	∞	0	−9	61
H	∞	12	−94	10	10	18	∞	∞	0	∞
I	−4	40	∞	76	31	81	−61	47	∞	0

Cene mitnin med posameznimi kraji so v butalskih tolarjih (BUT) in so cela števila. V primeru, da ni možno neposredno iz enega kraja v drugega, je to označeno z ∞. Posebej je označen kraj županovega bivališča $\boxed{\check{Z}}$.

Peter je dobil posebno nalogo, naj naračuna županu, koliko ga stane pot od njegovega doma do vsakega drugega kraja v Butalah. Spomnil se je, da bi v ta namen lahko uporabil Dijkstrov algoritem, a je kmalu spoznal, da ne bo šlo zaradi negativnih mitnin. Nekaj časa je brskal po spletu in našel Johnsonov algoritem, ki je namenjen iskanju najcenejših poti med vsemi pari krajev, a se ga je odločil prirediti:

1. Najprej dopolnimo zemljevid krajev tako, da dodamo nov kraj X, za katerega velja, da je mitnina od njega do vseh ostalih krajev 0 BUT.
2. Na spremenjenem zemljevidu s pomočjo Bellman–Fordovega algoritma naračunamo cene $c(k)$ najcenejših poti od X do vseh ostalih krajev $k \in \{A,B,C,\check{Z},D,E,F,G,H,I\}$ v Butalah.
3. Sledi prevrednotenje mitnin za potrebe iskanja najcenejše poti in sicer višina mitnine med krajema k in l se spremeni iz $m(k,l)$ v zgornji preglednici na $m(k,l)+c(k)-c(l)$. Odstrani se še kraj X.

Na tako popravljenem zemljevidu lahko Peter uporabi Dijkstrin algoritem za izračun najcenejših poti od županovega bivališča $\boxed{\check{Z}}$.

Ali je Petrov pristop **pravilen**?

A) Da, je, saj pretvori problem v obliko, ki omogoča uporabo Dijkstrinega algoritma.

B) Ne, ni, ker je Johnsonov algoritem namenjen reševanju drugega problema.

C) Morda je pravilen, a ni smiseln, saj bi lahko zemljevid popravil tako, da bi vsem mitninam prištel takšno vrednost, da bi postale nenegativne — na primer v primeru Butal 94.

D) Ne ni, saj bi moral prekiniti izvajanje, če bi v drugem koraku zaznal negativen cikel.

People from Butale are strange people. Sometimes they are completely normal, but sometimes they quarrel with their mind. So they started charging tolls on Butale roads. But they wouldn't be Butalians if they didn't charge a negative toll on some roads — in other words, they even paid the person who drove on such a road. To make everything more transparent, they prepared a table in which they entered the toll amount between individual Butale towns:

	A	B	C	$\boxed{\check{Z}}$	D	E	F	G	H	I
A	0	−94	12	19	72	81	∞	∞	7	−64
B	∞	0	83	∞	∞	55	∞	46	9	∞
C	−12	−57	0	37	36	28	86	95	36	97
$\boxed{\check{Z}}$	−86	−50	96	0	79	∞	−86	∞	∞	∞
D	−21	∞	∞	∞	0	∞	∞	41	67	54
E	−61	−1	∞	65	∞	0	41	∞	53	∞
F	∞	78	∞	71	28	88	0	6	−47	100
G	∞	−45	∞	16	53	∞	∞	0	−9	61
H	∞	12	−94	10	10	18	∞	∞	0	∞
I	−4	40	∞	76	31	81	−61	47	∞	0

The toll prices between individual locations are in Butale tolars (BUT) and are integer numbers. In case it is not possible to go directly from one location to another, this is indicated by ∞. The mayor's home town $\boxed{\check{Z}}$ is specially marked.

Peter was given a special task: to calculate for the mayor how much it costs to travel from his home to every other place in Butale. He remembered that he could use Dijkstra's algorithm for this purpose, but he soon realized that it wouldn't work because of negative tolls. He spent some time surfing the Internet and found Johnson's algorithm, which is designed to find the cheapest routes between all pairs of places, but he decided to modify it for his needs:

1. First, let's complete the map of places by adding a new place X, for which the toll from it to all other places is 0 BUT.

2. On the modified map, using the Bellman–Ford algorithm, compute the prices $c(k)$ of the cheapest routes from X to all other locations $k \in \{A,B,C,\check{Z},D,E,F,G,H,I\}$ in Butale.

3. Next, the tolls are revalued for the purpose of finding the cheapest routein the original map. The toll between locations k and l is changed from $m(k,l)$ in the table above to $m(k,l)+c(k)-c(l)$. The location X is removed.

On a so modified map, Peter can use Dijkstra's algorithm to compute the cheapest routes from the mayor's residence $\boxed{\check{Z}}$.

Is Peter's approach even **correct**?

A) Yes, it is, because it transforms the problem into a form that allows the use of Dijkstra's algorithm.

B) No, it is not, because Johnson's algorithm is designed to solve a different problem.

C) It may be correct, but it does not make sense, because it could fix the map by adding a value to all the tolls so that they become non-negative — for example, in the case of Butale 94.

D) No, it is not, because it would have to stop execution if it detected a negative cycle in the second step.

8. naloga

Dani naj bosta funkciji $T, S: \mathbb{N} \to \mathbb{N}$, ki zadoščata rekurzivnima zvezama

$$T(n) = \begin{cases} 2 \cdot T(\lfloor n/2 \rfloor) + 4n & \text{če } n \geq 10, \\ 5 & \text{če } n < 10; \end{cases}$$

$$S(n) = \begin{cases} S(n-1) + 4n & \text{če } n \geq 10, \\ 5 & \text{če } n < 10. \end{cases}$$

Katera izmed spodnjih izjav **je resnična**?

A) $T(n) = O(n \log n)$ in $S(n) = O(n \log n)$.

B) $T(n) = O(n^2)$, $T(n)$ ni v $O(n \log n)$, in $S(n) = O(n \log n)$.

C) $T(n) = O(n \log n)$, $S(n) = O(n^2)$, in $S(n)$ ni v $O(n \log n)$.

D) $T(n) = O(n^2)$, $T(n)$ ni v $O(n \log n)$, $S(n) = O(n^2)$, in $S(n)$ ni v $O(n \log n)$.

Consider the following recurrences for functions $T, S: \mathbb{N} \to \mathbb{N}$

$$T(n) = \begin{cases} 2 \cdot T(\lfloor n/2 \rfloor) + 4n & \text{if } n \geq 10, \\ 5 & \text{if } n < 10; \end{cases}$$

$$S(n) = \begin{cases} S(n-1) + 4n & \text{if } n \geq 10, \\ 5 & \text{if } n < 10. \end{cases}$$

Which of the following statements **is true**?

A) $T(n) = O(n \log n)$ and $S(n) = O(n \log n)$.

B) $T(n) = O(n^2)$, $T(n)$ is not in $O(n \log n)$, and $S(n) = O(n \log n)$.

C) $T(n) = O(n \log n)$, $S(n) = O(n^2)$, and $S(n)$ is not in $O(n \log n)$.

D) $T(n) = O(n^2)$, $T(n)$ is not in $O(n \log n)$, $S(n) = O(n^2)$, and $S(n)$ is not in $O(n \log n)$.