Korespondenční Seminář z Programování

ZAČÁTEČNICKÁ KATEGORIE

34. ročník **KSP-Z** **Duben 2022**

Řešení čtvrté série začátečnické kategorie 34. ročníku KSP

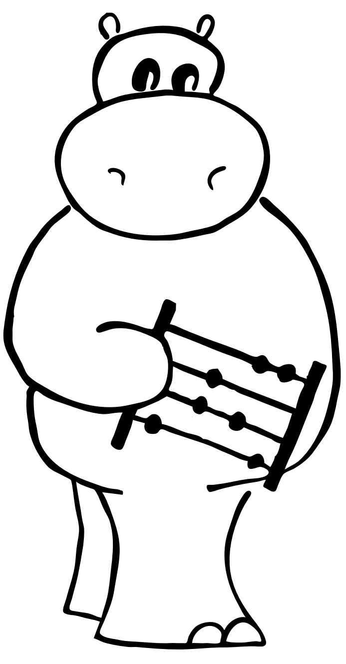
34-Z4-1 Halleyova kometa

Úlohy se snadno můžeme chopit hrubou silou. Budeme postupně procházet vstupní seznam lidí a pro každého spočítáme následující:

```
opakuj pro každý rok mezi narozením
      a úmrtím člověka:
  pokud doba od příletu je dělitelná periodou:
    tento člověk kometu mohl spatřit
    přejdi na dalšího člověka
```

První vstup takto nejspíš vyřešíme, ale pro zákeřnější vstupy se náš procesor překně zapotí. Zkoušet všechny roky je příliš mnoho práce.

Můžeme zkusit opačný postup. Zaměříme se na roky, kdy kometa přiletěla, a poté zjistíme, jestli to bylo za života dané osoby. Dokonce nám stačí pro každou osobu prozkoumat jenom jeden přílet komety: ten poslední před tím, než daný člověk zemřel. Pokud nastal po roku narození, daný člověk kometu mohl vidět; pokud před ním, pak nemohl.



Zbývá vyřešit, jak najít rok posledního příletu. Podobný problém jsme již řešili v úloze 34-Z1-1 Letopočty. Je-li u rok úmrtí, o rok prvního objevení komety a p její perioda, rok jejího posledního příletu před úmrtím dané osoby bude $p \cdot \lfloor (u-o)/p \rfloor + o$.

Program (Python 3):
`http://ksp.mff.cuni.cz/viz/34-Z4-1.py`

Úlohu připravili: Vojta Káně, Ondra Sladký

34-Z4-2 Letadla na ranveji

Představme si, že letadla přiřazujeme k destinacím od nejvzdálenější destinace.

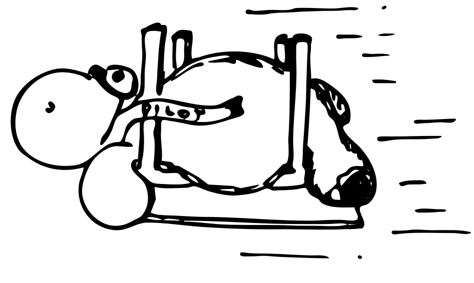
Počet letadel, která mohou letět nejvzdálenější trať, můžeme spočítat snadno. Jsou to ta letadla, která mají větší dolet, než je tato vzdálenost. Označme tento počet l_1.

Podobným způsobem můžeme spočítat i l_2 – kolik letadel může letět druhou nejdelší trasu. Nyní by nás zajímalo, kolika způsoby můžeme letadla přidělit ke dvěma nejdelším trasám. Víme, že k nejdelší trase můžeme přiřadit l_1 letadel. Jakmile jedno přiřadíme, dále jej již nemůžeme použít, máme tedy k dispozici o letadlo méně. Jelikož každé letadlo, které je schopné uletět nejdelší trasu, určitě může letět druhou nejdelší trasu, ke druhé trase můžeme přiřadit pouze $l_2 - 1$ letadel. Volbu prvního a druhého letadla můžeme kombinovat, celkově se tak jedná o $l_1 \cdot (l_2 - 1)$ možností.

V této úvaze můžeme pokračovat. Jakmile již máme přiřazena letadla k nejdelším dvěma trasám, ať už jsme to udělati jakkoliv, na třetí nejdelší nám zbývá $l_3 - 2$, protože l_3 je jich schopno trasu uletět, ale 2 z nich jsme již vždy použili. Je tedy $l_1 \cdot (l_2 - 1) \cdot (l_3 - 2)$ možností, jak letadla rozmístit na nejdelší 3 trasy.

Jakmile tento postup uděláme pro všechny destinace, dostaneme, že celkový počet možností je $l_1 \cdot (l_2 - 1) \cdot (l_3 - 2) \cdot \ldots \cdot (l_n - (n-1))$, což můžeme pomocí velkého součinu napsat kompaktně jako $\prod_{i=1}^{n} l_i - (i-1)$, kde l_i je počet letadel, která uletí i-tou nejdelší trasu.

Všimněme si, že tento vzoreček bude fungovat i pokud žádné řešení neexistuje. Na prvním místě, kdy nejde žádné letadlo přiřadit k dané trase, se nám objeví nula a tedy i celkový výsledek bude nulový.



Nyní zbývá zjistit všechny hodnoty l_i. Jakmile je budeme mít, umíme počet přiřazení spočítat pomocí vzorečku výše v lineárním čase. Musíme ovšem během výpočtu po každém násobení brát zbytek po dělení $10^9 + 7$, aby nám výsledek nepřetekl, popřípadě se nám nezpomalila aritmetika.

To zvládneme následovně. Setřídíme si letadla i trasy od největšího po nejmenší. Nyní budeme zjišťovat l_1. Budeme procházet od začátku pole s letadly, dokud nenarazíme na nejvýkonější letadlo, kterou první trasu neurazí. Pak víme, že všechna předchozí trasu k nejvzdálenější destinaci uletí, zbylá nikoliv. Z toho již víme l_1. Stejně tak můžeme postupovat i pro další trasy, jen si stačí uvědomit, že nemusíme začínat od začátku, nýbrž od místa, kde jsme skončili u předchozí trasy, jelikož letadla předchozí letadla uletí i delší trasy. Celkově tak zvládneme získat všechna l_i jedním průchodem oběma poli.

Nejpomalejší operací bylo řazení, celková složitost je tak $\mathcal{O}(n \log n)$.

Nejvíce prostoru zabírají vstupní pole, popř. pomocné pole hodnot l_i, celková paměťová složitost je tak $\mathcal{O}(n)$.

Program (Python 3):
`http://ksp.mff.cuni.cz/viz/34-Z4-2.py`

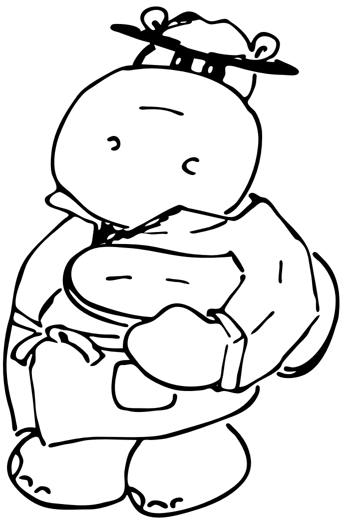
Úlohu připravili: Jirka Kalvoda, Ondra Sladký

34-Z4-3 Dozorci v bludišti

Na první pohled se může zdát, že optimálním řešením je dozorce simulovat přesně podle zadání. Stejně musíme celou mapu přečíst ze vstupu. Není to však pravda. Počet kroků simulace může být mnohem větší než počet políček na mapě, speciálně $\mathcal{O}(PMN)$.[1]

Naštěstí má úloha jednu důležitou vlastnost – dozorci spolu nijak neinteragují. Můžeme je tedy simulovat každého zvlášť a pamatovat si, která políčka skončila obsazena.

Navíc si můžeme uvědomit, že je-li kroků simulace výrazně více než políček mapy, musí dozorce navštívit nějaké políčko ve stejném směru vícekrát – začne chodit po cyklu. Tím nám velká část simulace odpadne: pokud je délka cyklu k, můžeme poté, co se dozorce na cyklus dostane, pokaždé k kroků přeskočit, protože by se dozorce jen vrátil na to samé místo. Chceme tudíž tento cyklus najít.



Hledání cyklu je známý problém, který lze řešit v lineárním čase vzhledem k součtu jeho délky a délky počáteční cesty. My zde popíšeme jeden algoritmus, další můžete najít např. na Wikipedii.[2]

Pořídíme si želvu a zajíce, oba dva umístíme na počáteční pozici dozorce. Oba chodí po mapě stejně jako dozorce, zajíc za každý krok želvy udělá kroky dva. Jakmile želva přejde počáteční cestu a oba se tak dostanou na cyklus, začne zajíc honit želvu. S každým krokem zajíc želvu o jedno políčko dožene, v lineárním čase s délkou cyklu se tedy setkají.

Dejme tomu, že se zajíc a želva setkali po k krocích na políčku P. To znamená, že zajíc urazil o k políček více než želva. Tuto vzdálenost nemohl nabrat jinde než v cyklu, takže k je násobkem délky cyklu (ne nutně 1-násobkem, uvažme dlouhou cestu ke krátkému cyklu). Délku cyklu pak zjistíme podle toho, kolik kroků potrvá želvě znovu navštívit políčko P.

Pro naše účely však stačí znát k. Po prvních k krocích dojde dozorce na políčko P. Po každých dalších k krocích se na něj vrátí. Stačí nám tedy najít zbytek po dělení celkového počtu kroků číslem k a odsimulovat tolik kroků od políčka P, tím najdeme jeho pozici na konci simulace.

Ještě by se mohlo stát, že kroků simulace je tak málo, že se zajíc s želvou nestihnou potkat. To ale nevadí, pak rovnou víme, že by dozorce skončil tam, kde želva.

Jak dlouho tato vylepšená simulace potrvá? Pro každého dozorce zjistíme násobek délky cyklu k, provedeme zbytek po dělení a provedeme až k kroků simulace. Vše zvládneme v čase $\mathcal{O}(k)$. Jelikož k je lineární v počtu políček a na každém políčku může být dozorce, celkově strávíme čas $\mathcal{O}(N^2M^2)$.

Program (Python 3):
`http://ksp.mff.cuni.cz/viz/34-Z4-3.py`

Program (C):
`http://ksp.mff.cuni.cz/viz/34-Z4-3.c`

Úlohu připravili: Jirka Kalvoda, Vojta Káně

34-Z4-4 Kamínky

Dostali jsme kamínky na šachovnici $R \times S$. Chceme je dostat z počáteční pozice do cílové, přičemž smíme jenom rotovat řádky a sloupečky. Nebo máme říci, že to nejde. (Můžeme přitom předpokládat, že $R \leq S$ – jinak prostě šachovnici otočíme o 90 stupňů. Také si budeme představovat, že šachovnice je „cyklická“ – pod posledním řádkem leží první apod.)

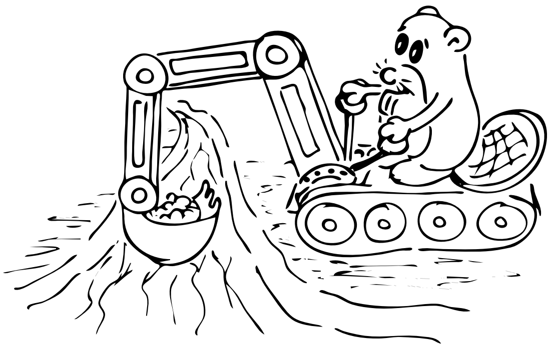
Začneme netradičně: zkusíme najít nějaký případ, kdy to nejde. Třeba pokud počáteční a cílová konfigurace obsahují různé počty kamínků :) Dobrá, to byl laciný trik, tak hledejme další neřešitelná zadání.

Zkusíme, jak se úloha chová pro malé šachovnice. Pro 1×1 a 1×2 to vždy jde. Pro 1×3 také: pokud tam je jeden kamínek, můžeme ho přesunout kamkoliv; pokud dva, můžeme si představit, že přesouváme jednu díru. Pro 1×4 se ale konfigurace „dva kamínky vedle sebe“ nedá zrotovat na „dva kamínky oddělené dírou“. Podobný příklad najdeme pro $1 \times S$, kdykoliv $S \geq 4$.

To je překvapivě všechno. Kdykoliv je R i S aspoň 2, řešitelné je každé zadání, které zachovává počet kamínků. Dokážeme to tak, že předvedeme, jak kterýkoliv kamínek přesunout do kterékoliv díry.

[1] To je dokonce exponenciální vzhledem k počtu znaků vstupu. Čas totiž závisí na *hodnotě* čísla, avšak jeho *desítkový zápis* ve vstupu je logaritmicky dlouhý.

[2] `https://en.wikipedia.org/wiki/Cycle_detection#Algorithms`



Nejprve to zkusíme pro šachovnice $2 \times S$ a případ, kdy je kamínek i díra v prvním řádku. Na chvíli předpokládejme, že v druhém řádku je aspoň jedna díra – té budeme říkat *skladiště.* Situaci sledujme na následujícím obrázku. Nejprve zrotujeme první řádek tak, aby se kamínek dostal nad skladiště. Pak rotací sloupečku zasuneme kamínek do skladiště. Následně zrotujeme první řádek, aby se nad skladiště dostala cílová díra. Poté rotací sloupečku dostaneme kamínek ze skladiště do díry. A nakonec zrotujeme první řádek do původní polohy.

1	○	2		3
4	5	6	7	

→

2		3	1	○
4	5	6	7	

→

2		3	1	
4	5	6	7	○

↓

1		2	○	3
4	5	6	7	

←

3	1		2	○
4	5	6	7	

←

3	1		2	
4	5	6	7	○

Všimněte si, že skladiště se na konci opět vyprázdní a všechna ostatní políčka v obou řádcích zůstanou nezměněna.

Dobře, ale co kdyby v druhém řádku byly samé kamínky? V tom případě si představíme, že místo kamínku v prvním řádku přesouváme díru :)

Takže umíme přesunout kamínek v rámci prvního řádku. Přesun v druhém řádku je stejný (nezapomeňte, že naše šachovnice je cyklická, takže všechny řádky se chovají stejně). Kdybychom chtěli kamínek přesunout v rámci sloupce, stačí zrotovat sloupec. A pokud ho chceme přesunout z prvního řádku na jinou pozici v druhém řádku, zrotujeme druhý řádek tak, aby se díra dostala pod kamínek, přesuneme kamínek v rámci sloupce, a pak zrotujeme druhý řádek zpátky.

Teď už umíme hru vyhrát na libovolné šachovnici $2 \times S$. Ukážeme, že podobný postup se dá použít na obecnou šachovnici $R \times S$, kdykoliv $R, S \geq 2$.

Začneme přesunem v rámci řádku. Místo jednoho skladiště budeme používat dvě: *sklípek,* což bude díra v řádku pod aktuálním, a *půdu* – políčko o dva řádky nad sklípkem. (Dokud existovaly jen dva řádky, sklípek a půda splývaly.)

Opět sledujme obrázek. Nejprve zrotujeme aktuální řádek, aby se přesouvaný kamínek dostal nad sklípek. Pak zrotujeme sloupec se sklípkem o 1 nahoru: kamínek se dostane na půdu a nahradí ho díra ze sklípku. Poté zrotujeme aktuální řádek, aby se nad sklípek dostala cílová díra. Zrotováním sloupce o 1 dolů se kamínek z půdy dostane na cílovou pozici a díra z této pozice do sklípku. Nakonec zrotujeme řádek zpět do původní polohy.

A	B	C	D	E
1	○	2		3
4	5	6	7	
P	R	S	T	Y

→

A	B	C	D	E
2		3	1	○
4	5	6	7	
P	R	S	T	Y

→

A	B	C	D	○
2		3	1	
4	5	6	7	Y
P	R	S	T	E

↓

A	B	C	D	E
1		2	○	3
4	5	6	7	
P	R	S	T	Y

←

A	B	C	D	E
3	1		2	○
4	5	6	7	
P	R	S	T	Y

←

A	B	C	D	○
3	1		2	
4	5	6	7	Y
P	R	S	T	E

Rozmyslime si, že všechny ostatní kamínky se přitom dostaly tam, kde byly na začátku. Situaci, kdy jsme ve „sklípkovém“ řádku nenašli žádnou díru, opět ošetříme přesouváním díry místo kamínku.

Přesun v rámci sloupce zařídíme tímtéž postupem, jen o 90 stupňů otočeným. A pokud chceme přesunout do jiného řádku i sloupce, nejprve zrotujeme cílový řádek, aby se cílové políčko dostalo do stejného sloupce jako kamínek. Pak přesuneme v rámci sloupce a nakonec cílový řádek zrotujeme zpátky.

Dokázali jsme tedy, že pro $R, S \geq 2$ je možné kterýkoliv kamínek přesunout do kterékoliv díry. Tím pádem z každého rozmístění kamínků lze udělat libovolné jiné se stejným počtem kamínků.

Úlohu připravili: Martin „Medvěd“ Mareš, Ondra Sladký

Výsledková listina čtvrté série začátečnické kategorie 34. ročníku KSP

	řešitel	*škola*	*ročník*	*sérií*	*Z4-1*	*Z4-2*	*Z4-3*	*Z4-4*	*série*	*celkem*
0.					8	12	12	12	44,0	176,0
1.	Zuzana Aubrechtová	GHeyrovPH	3	4	8	12	12	10	42,0	170,0
2.	Jáchym Kouba	GJŠkodyPŘ	2	4	8	12	12	11	43,0	163,2
3.	Adam Jahoda	GKepleraPH	3	6	8	12		12	32,0	156,0
4.	Jakub Hampl	GMělník	2	4	4	2	10	10	26,0	147,5
5.	Anna-Kristina Migel	GNAlejíPH	−1	4	8	4	10,7		22,7	145,7
6.	Kryštof Maxera	GJírovcČB	1	11	8	12			20,0	143,5
7.	Richard Tichý	SG Kladno	0	4	8	2			10,0	142,0
8.	Jakub Mikeš	GJŠkodyPŘ	4	7	8	12			20,0	139,0
9.	Ján Plachý	G VBN Prie	4	4	8		8		16,0	131,5
10.	Viktor Helmich	GTMannaPH	3	3	8	12	12	12	44,0	130,0
11.	Alexandr Bihun	GJírovcČB	2	4	8	9	10		27,0	129,5
12.	Vojtěch Lančarič	SPŠG Třebešín	3	3	8	12	10	10	40,0	128,0
13.	Kryštof Latka	PORG Krč	4	7	8				8,0	126,5
14.	Lukáš Létal	GJŠkodyPŘ	3	9	8	12			20,0	122,0
15.	Václav Kouřil	GTachov	4	4	8	12	2		22,0	121,0
16.	Vít Olšovec	GPřípotoPH	0	4	8	12	8		28,0	120,0
17.	Vladimír Sklenár	GTerVans	2	4	8	2	8		18,0	114,0
18.	Lukáš Linek	GOpatovPHA	−2	3					0,0	112,0
19.	Oto Skýpala	GJŠkodyPŘ	−2	7	8	12	8		28,0	110,5
20.	Matúš Púll	GZborovPH	2	4	8	2	8		18,0	108,0
21.	Jakub Kopčil	GMikulášPL	3	7	8	12		5	25,0	107,3
22.	Matěj Hošek	GVolgogrOS	0	5	8	0			8,0	103,0
23.–24.	Jan Černohorský	G Brandýs	4	3	8	12	8		28,0	101,5
	Jakub Smolík	GEBenešeKL	4	7					0,0	101,5
25.	Bobur Toshtemirov	GMikulášPL	3	3	8	12	10		30,0	100,0
26.	Tomáš Janovec	GMnichHrad	4	9					0,0	94,5
27.	Viktor Číhal	SPŠSmíchov	2	5					0,0	93,5
28.	Daniel Šoltýs	GTřeKošice	4	7	8	0			8,0	91,0
29.	Jáchym Löwenhöffer	GEvolutionJM	1	4	8				8,0	88,3
30.–32.	Olga Cinková	ArcibisGPH	2	9	8				8,0	88,0
	Adam Kolník	SSŠVTPraha	3	7					0,0	88,0
	Šimon Šustek	G Brandýs	4	2					0,0	88,0
33.	Jan Prosecký	GNoMěsNMor	3	4	8				8,0	84,0
34.	Jakub Podskalský	SSŠVTPraha	2	4	1				1,0	82,0
35.	Jiří Kruchina	GČeskoliPH	4	4	8	12	8		28,0	81,0
36.	Tomáš Pražák	GJSeiferPH	1	5					0,0	80,0
37.–38.	Matúš Duchyňa	GGrössBA	3	3					0,0	79,0
	Ivan Trenčanský	GLSáru	3	3					0,0	79,0
39.	Nikolay Fomichev	SSŠVTPraha	3	3					0,0	78,0
40.	Svatava Šimečková	GJarošeBO	0	4					0,0	77,0
41.	Tadeáš Zíka	SPŠSmíchov	1	4	8	2			10,0	75,0
42.	Alexandra Seďová	GVídeňskBO	1	4	8	0			8,0	70,0
43.	David Pacák	G Brandýs	1	2					0,0	69,5
44.	Šimon Durda	PORG Ostrava	1	3	8	2			10,0	69,0
45.	Thomas Riedle	BRG APP	3	12					0,0	63,5
46.	Štěpán Fröde	GDobruška	2	2					0,0	61,0
47.	Matěj Strnad	SPŠJičín	1	9	0				0,0	57,0
48.	Jaromír Obitko	ZS6 Kladno	0	2					0,0	56,0
49.	Jan Straka	VOŠ Ždár	2	4	0				0,0	55,0
50.–51.	Jan Hlavsa	GMělník	4	7					0,0	54,0
	Kostia Kolomiiets	GVoděraPH	2	2	8		8		16,0	54,0
52.–53.	Petr Starý	GJírovcČB	0	4	8				8,0	53,0
	Jan Šuráň	GZborovPH	4	2					0,0	53,0
54.	Vít Mitáš	GPolička	0	3					0,0	52,0
55.	Dominik Dembinný	ZŠMR Kladno	−2	2	8	12			20,0	50,0

	řešitel	škola	ročník	sérií	Z4-1	Z4-2	Z4-3	Z4-4	série	celkem
56.	Michal Budai	G JGJ PH	−3	2					0,0	49,0
57.	Jáchym Tuma	G FrýdlNOs	1	4	8				8,0	48,0
58.	Ondřej Stupka	GVolgogrOS	2	2					0,0	47,0
59.	Honza Kocourek	ParkLane	2	3	8				8,0	46,0
60.	Jan Slíva	MensaG	1	1	8	12	12	12	44,0	44,0
61.	Erik Sabol	GČeskoliPH	2	10					0,0	43,0
62.–64.	Stanislav Kozák	G Holice	4	2					0,0	42,0
	Marek Maškarinec	SPŠEMasLI	1	4					0,0	42,0
	Michal Mík	SSŠVTPraha	1	3					0,0	42,0
65.–67.	Adam Kuča	PORG Krč	4	1					0,0	41,0
	Vojtěch Procházka	MensaG	1	1	8	12	10	11	41,0	41,0
	Vojtěch Venzara	GMělník	4	7					0,0	41,0
68.	Veronika Jůzková	MensaG	4	13	0				0,0	40,7
69.–71.	Radek Bláha	GČeskáČB	0	6					0,0	40,0
	Filip Neubauer	AkademGPH	2	1					0,0	40,0
	Michal Pavlíček	MendelGOP	4	1					0,0	40,0
72.–73.	Petr Hladík	GMikulášPL	4	7					0,0	38,0
	Miroslav Kolouch	GJírovcČB	2	2	8		12		20,0	38,0
74.	Alexander Mateides	GJirsíkaČB	3	5					0,0	36,0
75.	Ondřej Pupík	GRožnovPR	2	1	8	12	12	2	34,0	34,0
76.	Filip Šimek	GTurnov	3	3	4				4,0	32,3
77.	Samuel Dembinný	SPŠ Kladno	1	2					0,0	32,0
78.–82.	Pavel Altmann	GMikulášPL	3	8					0,0	30,0
	Kryštof Marek	SGPCE	2	4					0,0	30,0
	Jakub Nevařil	G UherBrod	4	13					0,0	30,0
	Nikol Poláková	GMetodovaBA	3	1					0,0	30,0
	Vojtěch Skyba	G UherBrod	4	4					0,0	30,0
83.–84.	Kateřina Doubková	GNAlejíPH	3	1	8	12	8		28,0	28,0
	Milan Savickij	SPŠSmíchov	2	1					0,0	28,0
85.	Albert Bakoč	GZborovPH	1	2	8		10		18,0	22,0
86.	Jakub Štefan	GMělník	3	1	8	12			20,0	20,0
87.	Šimon Hanák	CMG Brno	−1	3	1				1,0	19,0
88.–90.	Viktor Čubík	G UherBrod	4	2					0,0	18,0
	Petr Kroča	G UherBrod	1	8					0,0	18,0
	Lída Pavelková	GPatočkyPH	2	1					0,0	18,0
91.–92.	Matěj Kříž	GDašickáPA	4	1					0,0	16,0
	Arnošt Polák	PORG Krč	4	2					0,0	16,0
93.	Patrik Prinz	GJŠkodyPŘ	3	1	8	2	4		14,0	14,0
94.–95.	Robin Kovar	GPŠ Praha	0	1					0,0	12,0
	Lenka Poljaková	GJŠkodyPŘ	2	2	4				4,0	12,0
96.–97.	Adam Bureš	SPŠ Přerov	2	2					0,0	10,0
	Martin Müller	GZborovPH	1	1	8	2			10,0	10,0
98.	Štěpán Remeš	GZborovPH	2	1					0,0	9,0
99.–109.	Filip Cába	GEBenešeKL	1	1	8	0			8,0	8,0
	Adam Húšťava	EupSchoolLux	4	15					0,0	8,0
	Zara Karakaya	TAPoprad	4	1					0,0	8,0
	Jan Klokan	SPŠChom	1	1					0,0	8,0
	Luka Králík	GArc	1	1	8				8,0	8,0
	Ota Macourek	GZborovPH	2	2	0				0,0	8,0
	Tomáš Plášek	GZborovPH	1	1	8				8,0	8,0
	Lucian Poljak	GJŠkodyPŘ	0	1	8				8,0	8,0
	Karel Procházka	GPBystrica	4	1	8				8,0	8,0
	Jozef Remiš	G Bilíkova	3	2	0				0,0	8,0
	Marek Švajda	G UherBrod	4	1					0,0	8,0
110.	Roman Fiala	GChomutov	4	1					0,0	7,0
111.	Marek Plachý	GJatečníÚL	3	1					0,0	6,0

	řešitel	*škola*	*ročník*	*sérií*	*Z4-1*	*Z4-2*	*Z4-3*	*Z4-4*	*série*	*celkem*
112.	Adam Jirásek	G Brandýs	2	1	5				5,0	5,0
113.–115.	Jáchym Hájek	GBNěmcovHK	−1	4					0,0	4,0
	Robert Klimt	GDobříš	2	1					0,0	4,0
	Matyas Oliva	G UherBrod	4	2					0,0	4,0
116.–119.	Radim Guichen	GJírovcČB	0	1					0,0	2,0
	Janek Hlavatý	GJirsíkaČB	3	22					0,0	2,0
	Jan Koška	GJírovcČB	2	8					0,0	2,0
	Kateřina Vomelová	GÚstavníPH	2	1					0,0	2,0
120.–122.	Yahor Herashchanka	GTurnov	1	5					0,0	1,0
	Mikuláš Jandík	GZborovPH	1	1	0,3	0,7			1,0	1,0
	Jan Kopal	SPŠEMasLI	2	1	1				1,0	1,0
123.	Milan Vencl	GČesLípa	2	1	0,3				0,3	0,3