

# Najdi si svou cestu

Interaktivní pomůcka pro výuku  
informatiky konstruktivistickou metodou.

PRŮVODCE PRO UČITELE

## Gymnázium a SŠ

Plán tří hodin a ověření, karty aktivit včetně srovnání algoritmů a vše, co potřebujete k přípravě.

**Stačí přečíst první dvě strany.** Zbytek průvodce je příručka: plán hodin a krátké karty aktivit, ke kterým se vrátíte, až je budete potřebovat. Listy, ověření a klíč řešení jsou v samostatném souboru ke kopírování, ovládání apletu v Krátkém návodu k apletu. Informatické vzdělání nepotřebujete.

### O co jde

Studenti nejdříve sami prozkoumají graf, který nevidí celý, a hledají vlastní pravidlo průchodu. Teprve potom ho porovnají s pojmenovanými algoritmy (BFS, DFS, Dijkstra), které v apletu krokují a předpovídají jejich další krok. Převádějí graf mezi zápisy a posuzují jejich nároky.

### Co se žáci mohou naučit

- převádět graf mezi nákresem, seznamem sousedů, seznamem hran a maticí a hodnotit nároky zápisů
- předpovědět běh BFS, DFS a Dijkstrova algoritmu a vysvětlit, co který zaručuje
- rozšíření: minimální kostra (Kruskal, Jarník, Borůvka) a izomorfismus

### Kolik času a co potřebujete

|                 |                                                                                                             |
|-----------------|-------------------------------------------------------------------------------------------------------------|
| Čas             | Minimálně 3 vyučovací hodiny + ověření ve 4. hodině (45 minut); rozšíření do dalších hodin nebo do semináře |
| Technika        | počítač pro studenta                                                                                        |
| Profil v apletu | „Gymnázium – čtyřleté a vyšší stupeň“ (odborné názvosloví)                                                  |
| Listy           | 5A–5C a ověření 5; rozšíření 5D–5E                                                                          |

### Pět zásad, které stačí znát

- **Nejdříve zkouší žák, pak teprve vysvětlujete.** Neradíte postup, ptáte se: „Jak víš, kde už jsi byl?“ „Podle čeho je ta cesta nejlepší?“
- **Obrázek sítě není mapa.** Když žák řekne „je to blízko, protože je to na obrázku vedle“, je to příležitost k rozhovoru, ne chyba k opravě.
- **Rychlost se nehodnotí.** Pomalejší, ale promyšlený postup je lepší než rychlé klikání.
- **Pojmenovaný algoritmus až po vlastním hledání.** I na gymnáziu student nejdříve hledá vlastní pravidlo; BFS, DFS nebo Dijkstru krokuje teprve potom a porovnává je se svým postupem.
- **Neprozrazujte strukturu skrytého grafu** a vždy se ptejte, podle jakého kritéria student cesty porovnává (počet hran, nebo součet ohodnocení).



## Slovníček – víc slov nepotřebujete

| Slovo          | Co znamená                                                             |
|----------------|------------------------------------------------------------------------|
| graf           | Vrcholy a hrany; náčrt nic neurčuje.                                   |
| vrchol, hrana  | Místo a spojení; orientovaná hrana vede jen jedním směrem.             |
| stupeň vrcholu | Počet hran, které z vrcholu vedou.                                     |
| ohodnocení     | Číslo u hrany (v běžném názvosloví apletu „cena“).                     |
| délka cesty    | Součet ohodnocení. Nejkratší cesta = nejmenší součet, ne nejméně hran. |

Profil nastaví odborné názvosloví sám. Pozor na rozdíl: v literatuře je „nejkratší“ cesta ta s nejmenším součtem ohodnocení – aplet na to studenty upozorní.

## Jak aplet ovládat

**Krátký návod k apletu** (společný pro všechny školy, tři strany) popisuje zadání aktivity v pěti krocích, co žák v apletu uvidí, odevzdání a přehled třídy. Pro tuto úroveň v editoru zvolte profil „Gymnázium – čtyřleté a vyšší stupeň“.

**Pracovní listy jsou volitelné.** Scénáře nabízejí jejich využití, ale hodinu můžete vést pouze s interaktivními aktivitami apletu. Když list vynecháte, věnujte jeho čas druhému pokusu v apletu nebo společnému porovnání postupů.



SCÉNÁŘ S5 · GYMNAZIUM A SŠ

# Graf jako model a algoritmy na grafech

Sekvence tří hodin + ověření (4. hodina); rozšíření pro seminář.

|                         |                                                                                                                                                                                                                                                                                                    |
|-------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Cíl</b>              | Student převádí graf mezi reprezentacemi a hodnotí jejich nároky, předpoví běh BFS, DFS a Dijkstrova algoritmu a zdůvodní, co který zaručuje.                                                                                                                                                      |
| <b>RVP G</b>            | 5.8.1 Informatika – Data, informace a modelování (převod mezi modely, chyby modelu; učivo graf, orientovaný a ohodnocený graf); Algoritmizace a programování (vysvětlení algoritmu, nároky, oprava chyby).                                                                                         |
| <b>Pomůcky</b>          | Počítače; listy 5A–5C, pro seminář 5D–5E; ověření 5 (45 min).                                                                                                                                                                                                                                      |
| <b>Nastavení apletu</b> | Profil Gymnázium (odborné názvosloví). H1: průzkum skrytého grafu, pak Seznam sousedů, Seznam hran, Matice 0/1 (najít chybu). H2: Krokování BFS a DFS. H3: Krokování Dijkstra na grafu se shodou vzdáleností. Seminář: Nejlevnější propojení (volně → Kruskal, Jarník, Borůvka), Stejná síť jinak. |

## H1 – reprezentace a jejich nároky

| Fáze         | Min | Co dělají žáci                                          | Co dělá učitel                   | Materiál |
|--------------|-----|---------------------------------------------------------|----------------------------------|----------|
| Průzkum      | 10  | Projdou skrytý graf a zapíšou ho vlastním způsobem.     | Sbírá různé zápisy.              | aplet    |
| Reprezentace | 15  | Seznam sousedů, hran, matice v apletu; matice s chybou. | Vede k vlastnostem matice.       | aplet    |
| List 5A      | 20  | Převody, stupně, orientovaná matice, nároky.            | Diskuse: kdy matice, kdy seznam. | list 5A  |

## H2 – BFS a DFS

| Fáze             | Min | Co dělají žáci                                          | Co dělá učitel                               | Materiál |
|------------------|-----|---------------------------------------------------------|----------------------------------------------|----------|
| Vlastní pravidlo | 8   | Pravidlo systematického průchodu.                       | Nechá porovnat pravidla.                     | —        |
| Krokování        | 22  | BFS a DFS s tipováním.                                  | Vysvětlí frontu a zásobník; řazení abecedně. | aplet    |
| List 5B          | 15  | Vrstvy, zásobník, předpověď, oprava chybného algoritmu. | —                                            | list 5B  |

## H3 – Dijkstrův algoritmus

| Fáze      | Min | Co dělají žáci                                           | Co dělá učitel                        | Materiál |
|-----------|-----|----------------------------------------------------------|---------------------------------------|----------|
| Problém   | 5   | Proč BFS nestačí na ohodnocený graf?                     | Protipříklad na tabuli.               | —        |
| Krokování | 15  | Dijkstra s tipováním; shoda vzdáleností.                 | Ptá se na nejmenší známou vzdálenost. | aplet    |
| List 5C   | 20  | Tabulka běhu, rekonstrukce cesty, záporná hrana, nároky. | —                                     | list 5C  |



| Fáze    | Min | Co dělají žáci                   | Co dělá učitel | Materiál |
|---------|-----|----------------------------------|----------------|----------|
| Shrnutí | 5   | BFS × Dijkstra: co minimalizují. | —              | —        |

## Diferenciace

| Podpora                                                                                        | Výzva                                                                                                                               |
|------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|
| Menší grafy (6 vrcholů); u DFS rozdat tabulku zásobníku předem; matice nejdříve $4 \times 4$ . | Seminář: minimální kostra třemi algoritmy (5D), izomorfismus a invarianty (5E); implementace BFS v Pythonu a srovnání s krokováním. |

## Podle čeho poznáte, že je cíl splněn

- Student bez chyby převede graf mezi třemi reprezentacemi a odhalí chybu v matici.
- Předpoví pořadí BFS i DFS podle pravidla apletu a vysvětlí rozdíl.
- Zdůvodní, proč Dijkstra a ne BFS, a ví, kdy Dijkstra selže.
- Ověření 5: aspoň 60 % základu.

## Na co si dát pozor

- DFS v apletu jde nejdříve do abecedně posledního souseda – vysvětlíte předem.
- Pojmenované algoritmy jsou v RVP G volbou školy; ukotvíte je v ŠVP.



## PŘÍRUČKA

# Karty aktivit

Ke každé aktivitě: co žák dělá, jak ji nastavit, na co se ptát a co může znamenat, když se žák zasekne.

## Průzkum grafu

|                      |                                                                                                                                                                                                                                                     |
|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Co žák dělá</b>   | Student prochází graf, který nevidí celý, a hledá vlastní pravidlo průchodu. Je to vstup: bez vlastní zkušenosti nemá krokování algoritmů o co se opřít.                                                                                            |
| <b>Jak nastavit</b>  | <ul style="list-style-type: none"> <li>Profil Gymnázium nastaví odborné názvosloví; síť sám neskrývá – pro průzkum zvolte „skrytá“.</li> <li>Graf s 8–10 vrcholy, s kružnicí a slepou větví; pro diskusi o kritériu rozdílná ohodnocení.</li> </ul> |
| <b>Na co se ptát</b> | <ul style="list-style-type: none"> <li>Jaké pravidlo jsi použil/a a zaručí, že projdeš všechno?</li> <li>Je cesta s nejméně hranami zároveň nejkratší?</li> </ul>                                                                                   |

| Když vidíte...                               | Co to může znamenat                                  | Zkuste...                                    |
|----------------------------------------------|------------------------------------------------------|----------------------------------------------|
| Student opakovaně vstupuje do týchž vrcholů. | Možná nemá záznam navštívených vrcholů ani pravidlo. | Zeptejte se, jak by poznal, že už všude byl. |
| Neví, kam kliknout nebo jak se vrátit.       | Spíš potíží s ovládáním než s učivem.                | Pomozte hned; v pilotu si to poznamenejte.   |

## Reprezentace grafu (Seznam sousedů, Seznam hran, Matice sousednosti)

|                      |                                                                                                                                                                                                                                                                                                         |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Co žák dělá</b>   | Student zapisuje týž graf třemi způsoby a v režimu „najdi chybu“ hledá chyby v cizím zápisu.                                                                                                                                                                                                            |
| <b>Jak nastavit</b>  | <ul style="list-style-type: none"> <li>Nejdřív vyplnit, potom „najít chybu“ se 2 chybami a zpětnou vazbou „jen kolik je správně“.</li> <li>Seznam hran bez předvyplněných vrcholů.</li> <li>Matice: nejdřív 0/1; v matici ohodnocení je 0 platná hodnota a „—“ znamená, že hrana neexistuje.</li> </ul> |
| <b>Na co se ptát</b> | <ul style="list-style-type: none"> <li>Kdy se vyplatí matice a kdy seznam sousedů?</li> <li>Co znamená, když matice není symetrická?</li> </ul>                                                                                                                                                         |

| Když vidíte...                         | Co to může znamenat                                                       | Zkuste...                                        |
|----------------------------------------|---------------------------------------------------------------------------|--------------------------------------------------|
| Matice není symetrická.                | Možná zapsal hranu jen jedním směrem – nebo považuje graf za orientovaný. | Zeptejte se na políčko [B, A] po zapsání [A, B]. |
| V seznamu hran chybí izolovaný vrchol. | Může považovat seznam hran za celý graf.                                  | „Kolik vrcholů má graf?“                         |

## Krokování algoritmu – BFS, DFS, Dijkstra

|                     |                                                                                                                                                                                                                                                           |
|---------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Co žák dělá</b>  | Student před každým krokem tipuje, který vrchol algoritmus zpracuje (u Dijkstry finalizuje) jako další; apilet tip ověří a ukáže frontu, zásobník nebo tabulku vzdáleností.                                                                               |
| <b>Jak nastavit</b> | <ul style="list-style-type: none"> <li>Až po vlastním průzkumu: BFS → DFS → Dijkstra.</li> <li>Pro Dijkstru graf s rozdílnými ohodnoceními a aspoň jednou shodou vzdáleností.</li> <li>Zobrazení řešení vypnete – tipování je podstatou úlohy.</li> </ul> |

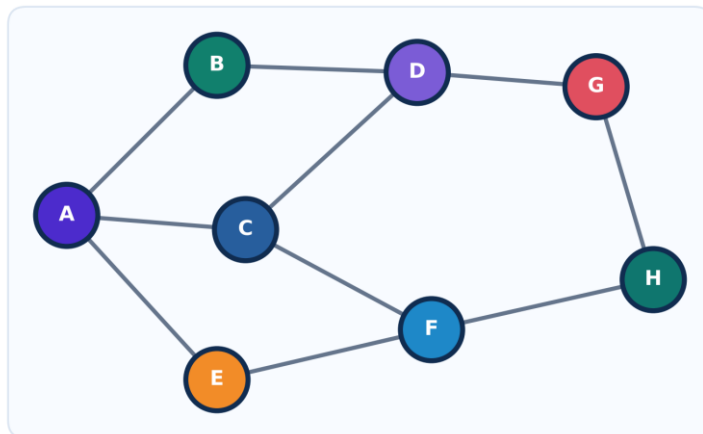


|                      |                                                                                                                                                         |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Na co se ptát</b> | <ul style="list-style-type: none"> <li>• Čím se liší tvoje pravidlo z průzkumu od BFS a DFS?</li> <li>• Proč BFS nestačí na ohodnocený graf?</li> </ul> |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------|

| Když vidíte...                                     | Co to může znamenat                                                                       | Zkuste...                                                          |
|----------------------------------------------------|-------------------------------------------------------------------------------------------|--------------------------------------------------------------------|
| Tipuje podle polohy na obrázku.                    | Možná zatím nepoužívá frontu ani zásobník.                                                | Nechte nahlas přečíst obsah fronty nebo zásobníku.                 |
| U DFS tvrdí, že „aplet dělá chybu“.                | Pravděpodobně čeká abecedně prvního souseda; aplet bere ze zásobníku abecedně posledního. | Projděte vkládání do zásobníku na tabuli.                          |
| U Dijkstry finalizuje vrchol, který objevil první. | Možná si plete Dijkstru s BFS.                                                            | Zeptejte se na nejmenší známou vzdálenost mezi nehotovými vrcholy. |

### Tři algoritmy: co dělají, čím se liší, kde se hodí

|                           | BFS – prohledávání do šířky                                                                                                                                    | DFS – prohledávání do hloubky                                                                                                                                                 | Dijkstrův algoritmus                                                                                                                                          |
|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Co dělá</b>            | Zpracovává vrcholy v pořadí, v jakém je objevil (fronta: dřív přidané jde dřív). Postupuje po vrstvách – nejdřív všichni sousedé startu, pak jejich sousedé... | Jde co nejdál jedním směrem; když nemá kam, vrací se k poslednímu vrcholu s nevyzkoušenou možností (zásobník: naposledy přidané jde dřív).                                    | Udrhuje nejmenší známou vzdálenost ke každému vrcholu. V každém kroku finalizuje nehotový vrchol s nejmenší vzdáleností a přepočítá vzdálenosti jeho sousedů. |
| <b>Co zaručuje</b>        | Najde cestu s nejmenším počtem hran ze startu ke každému vrcholu.                                                                                              | Projde vše dosažitelné ze startu; nejkratší cestu nezaručuje.                                                                                                                 | Najde cesty s nejmenším součtem ohodnocení – jen při nezáporných ohodnoceních.                                                                                |
| <b>Kde se hodí</b>        | Neohodnocený graf; nejméně přestupů; „kolik kroků je to odtud?“; šíření zprávy po vrstvách.                                                                    | Bludiště, dosažitelnost, hledání cyklu nebo komponent, prohledávání s návratem (backtracking).                                                                                | Ohodnocený graf: navigace, nejlevnější nebo nejrychlejší spojení, směřování v síti.                                                                           |
| <b>Pravidlo v apletu</b>  | Nové sousedy přidává na konec fronty abecedně.                                                                                                                 | Po zpracování vrcholu vloží jeho nenavštívené sousedy abecedně; zpracuje se vrchol nahoře – tedy abecedně <b>poslední</b> . Znovu objevený čekající vrchol se přesune nahoru. | Při shodě vzdáleností přijme kteroukoli rovnocennou volbu (sám volí abecedně první).                                                                          |
| <b>Typická chyba žáka</b> | Volí vrchol „blízko na obrázku“ místo podle fronty.                                                                                                            | Čeká abecedně prvního souseda; zapomene přesun čekajícího vrcholu nahoru.                                                                                                     | Finalizuje v pořadí objevení; zapomene vzdálenost snížit, když najde lepší cestu.                                                                             |



Na grafu z listu 5B: **BFS** od A zpracuje A, B, C, E, D, F, G, H (vrstvy 1: B, C, E · 2: D, F · 3: G, H). **DFS** podle pravidla apletu zpracuje A, E, F, H, G, D, C, B – jde nejdřív do abecedně posledního souseda. Rozdíl je dobré ukázat na tabuli dřív, než ho žáci objeví jako „chybu apletu“.

### Žákovský editor

|                      |                                                                                                                                    |
|----------------------|------------------------------------------------------------------------------------------------------------------------------------|
| <b>Co žák dělá</b>   | Student graf postaví: zachytí graf z průzkumu (aplet ho porovná s původním) nebo vytvoří model situace, i s orientovanými hranami. |
| <b>Jak nastavit</b>  | • Volná tvorba; orientovanou hranu označí políčkem „Jednosměrná“.                                                                  |
| <b>Na co se ptát</b> | • Je graf spolužáka jiný, nebo jen jinak nakreslený?                                                                               |

### Rozšíření – seminář nebo rychlejší studenti

Minimální kostra a izomorfismus nejsou v RVP G jmenovitě povinné.

#### Nejlevnější propojení (minimální kostra) (rozšíření)

|                      |                                                                                                                                                                                         |
|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Co žák dělá</b>   | Student propojí všechny vrcholy co nejlevněji – nejdřív vlastním pravidlem, potom krokuje Kruskalův, Jarníkův nebo Borůvkův algoritmus.                                                 |
| <b>Jak nastavit</b>  | • Vždy nejdřív volné hledání, pak krokování.<br>• Všechny tři algoritmy na stejném grafu – výsledná cena je stejná. Jarník začíná v počátečním vrcholu zadání – pro list 5D nastavte A. |
| <b>Na co se ptát</b> | • Proč má kostra vždy o jednu hranu méně, než je vrcholů?<br>• Je minimální kostra totéž co nejkratší cesty?                                                                            |

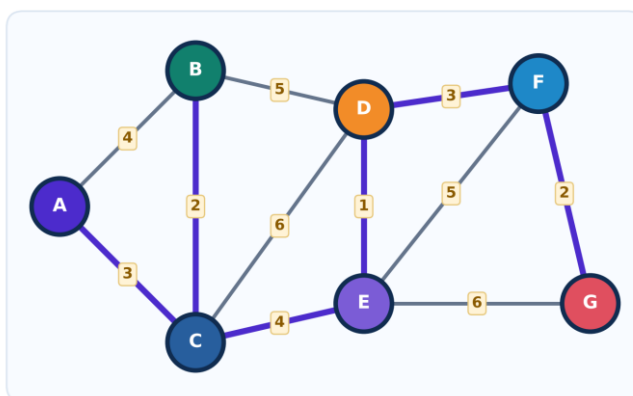
| Když vidíte...              | Co to může znamenat                       | Zkuste...                                                 |
|-----------------------------|-------------------------------------------|-----------------------------------------------------------|
| Výsledek obsahuje kružnici. | Možná nevidí, že jedna hrana je zbytečná. | „Kterou hranu můžeš vynechat, aby vše zůstalo propojené?“ |

### Tři algoritmy: co dělají, čím se liší, kde se hodí

|                | Kruskalův                                                                                             | Jarníkův (ve světě Primův)                                                                           | Borůvkův                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------|
| <b>Co dělá</b> | Seřadí všechna spojení od nejlevnějšího a bere je popořadě; přeskočí spojení, které by uzavřelo kruh. | Rozrůstá jeden celek od startu: vždy přidá nejlevnější spojení, které z už propojené části vede ven. | V každém kole si každá samostatná část vybere své nejlevnější spojení ven; všechna se přidají naráz. |



|                           | Kruskalův                                                   | Jarníkův (ve světě Primův)                                              | Borůvkův                                                                                  |
|---------------------------|-------------------------------------------------------------|-------------------------------------------------------------------------|-------------------------------------------------------------------------------------------|
| <b>Pohled</b>             | Globální – vidí seznam všech spojení.                       | Lokální – roste z jednoho místa.                                        | Souběžný – pracuje mnoho částí zároveň.                                                   |
| <b>Kde se hodí</b>        | Řídká síť zadaná seznamem hran; didakticky nejprůhlednější. | Síť budovaná od jednoho zdroje (vodovod, kabel od rozvodny); hustá síť. | Paralelní a distribuované výpočty; velké sítě.                                            |
| <b>Souvislosti</b>        | —                                                           | Vojtěch Jarník, 1930.                                                   | Otakar Borůvka, 1926 – úloha úsporné elektrifikace Moravy. Dobrá motivace pro české žáky. |
| <b>Typická chyba žáka</b> | Přidá levné spojení, které uzavře kruh.                     | Vybere nejlevnější spojení kdekoli, ne z rostoucí části.                | Nechá jednu část vybrat dvě spojení v témže kole.                                         |



Na grafu z listu 5D dají všechny tři algoritmy tutéž kostru za 15: Kruskal přidává D–E, B–C, F–G, A–C, D–F, C–E; Jarník od A přidává A–C, C–B, C–E, E–D, D–F, F–G; Borůvka v 1. kole A–C, B–C, D–E, F–G a ve 2. kole C–E, D–F. Kostra nezachovává nejkratší cesty: A–B je v grafu 4, v kostře 5.

### Stejná síť jinak (izomorfismus) (rozšíření)

|                      |                                                                                                                                                                                                                                                                                         |
|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Co žák dělá</b>   | Student rozhoduje, zda jsou dva různě nakreslené grafy tentýž graf, a hledá přiřazení vrcholů nebo invariant, ve kterém se liší.                                                                                                                                                        |
| <b>Jak nastavit</b>  | • Nejdřív „Přiřaď odpovídající místa“ (stejně grafy), pak „Rozhodni, jestli jsou sítě stejné“ s druhou sítí „Jiná“. Aplet ji pozmění ve spojeních, takže rozdíl často prozradí už stupně vrcholů. Grafy se stejnými stupni, které izomorfní nejsou, ukažte na grafech X a Y z listu 5E. |
| <b>Na co se ptát</b> | • Stačí stejné stupně vrcholů k tomu, aby byly grafy stejné?                                                                                                                                                                                                                            |

| Když vidíte...                       | Co to může znamenat                                    | Zkuste...                                     |
|--------------------------------------|--------------------------------------------------------|-----------------------------------------------|
| „Stejně stupně, takže stejné grafy.“ | Možná zatím nerozlišuje nutnou a postačující podmínku. | Ukažte protipříklad z listu 5E (grafy X a Y). |



PRO ŠVP A PLÁNOVÁNÍ

## Opora v RVP a cílová úroveň

Pro tematický plán, ŠVP nebo IVP – při běžné výuce číst nemusíte.

|                                  |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|----------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>RVP G – 5.8.1 Informatika</b> | <p><b>Data, informace a modelování:</b> „převede data z jednoho modelu do jiného; najde chyby daného modelu“; „formuluje problém a požadavky na jeho řešení; získává potřebné informace“. Učivo mj.: graf, vrcholy, hrany, orientovaný graf, ohodnocený graf.</p> <p><b>Algoritmizace a programování:</b> „vysvětlí daný algoritmus, program; určí, zda je daný postup algoritmem“; „ve vztahu k charakteru a velikosti vstupu hodnotí nároky algoritmů“; „ověří správnost, najde a opraví případnou chybu v algoritmu“.</p> <p>Konkrétní grafové algoritmy ani izomorfismus nejsou v RVP G jmenovitě povinné – jejich zařazení je rozhodnutím školy v ŠVP.</p> |
| <b>Kritická cesta</b>            | RVP G ji uvádí v učivu (Data, informace a modelování). Aplet ji neřeší, proto ji tato sada nezahrnuje; zařadte ji podle ŠVP vlastními materiály.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| <b>Cílová úroveň (základ)</b>    | Bezpečně převádí graf mezi nákresem, seznamem sousedů, seznamem hran a maticí, vyčte z nich stupně a počet hran a odhalí chybu; hodnotí paměťové nároky reprezentací; předpoví běh BFS, DFS a Dijkstrova algoritmu a zdůvodní, co který minimalizuje.                                                                                                                                                                                                                                                                                                                                                                                                           |
| <b>Rozšíření</b>                 | Minimální kostra (Kruskal, Jarník, Borůvka) a izomorfismus s argumentací pomocí invariantů.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |

### Zdroje

- RVP G – Informatika (2021): [digitalizace.rvp.cz/g/co-se-meni/nova-informatika](https://digitalizace.rvp.cz/g/co-se-meni/nova-informatika); úplné znění RVP G: [edu.gov.cz](https://edu.gov.cz)
- Borůvka, O. (1926); Jarník, V. (1930): O jistém problému minimálním. Práce Moravské přírodovědecké společnosti.
- Najdi si svou cestu – Krátký návod k apletu; Pracovní listy a ověření pro tuto úroveň (soubor Listy\_...).