

---

# Zadání soutěžní úlohy

## Kategorie programování webu

19. června 2021

Soutěž v programování – 34. ročník

Celostátní kolo 2020/2021

Jednotlivé části úlohy můžete řešit v libovolném pořadí a samozřejmě je nemusíte vyřešit všechny. Hodnotí se shoda se zadáním, funkčnost, dodržování webových standardů a přehlednost zdrojového kódu.

Na řešení úloh máte 4 hodiny čistého času.

Vaše řešení nahrávejte do cloudu přístupného na adrese <https://svp.proed.cz/>.

Přihlašovací jméno a heslo jste dostali emailem. Po přihlášení uvidíte složku odpovídající vašemu přihlašovacímu jménu (např. X123) a do ní řešení uložte. Pokud je řešení jedné úlohy složeno z více souborů, doporučujeme je nenahrávat samostatně, ale jako jeden ZIP archiv, jehož jméno odpovídá názvu úlohy (uloha-xyz.zip).

Vaše řešení musíte nahrát do 13:00, pozdější nahrání souborů nebude možné. Doporučujeme proto řešení nahrát s několikaminutovou rezervou. Pokud ve vaší kategorii řešíte více úloh, je vhodné každou úlohu nahrát hned, jak ji dokončíte.

Nahrávejte jak zdrojové kódy vašeho řešení, tak zkompilevané spustitelné soubory (pokud to ve vámi používaném programovacím jazyce dává smysl).

Stránku s řešením pojmenujte `index.html` a uložte v kódování UTF-8. Nezapomeňte nahrát všechny soubory (HTML, CSS, obrázky, zvuky, JavaScript).

Při zápisu HTML a CSS kódu dodržujte webové standardy. Řešení by mělo fungovat ve všech běžně dostupných prohlížečích.

Při řešení můžete používat frontendové technologie HTML, CSS a JavaScript. Všechny zdrojové kódy musíte napsat sami v průběhu soutěže. Řešení úlohy smí využívat pouze běžně dostupná API současných webových prohlížečů. Preprocesory zdrojového kódu (jako např. Babel, Typescript, Sass, Less, ...) jsou povoleny, odevzdávejte pak jak vámi napsaný vstupní kód, tak i vygenerovaný výsledek. Snažte se, aby vygenerovaný výsledek co nejlépe odpovídal zdrojovému souboru, neprovádějte minifikaci nebo obfuskaci.

Není povoleno používat knihovny, frameworky nebo kompilátory, které vyžadují runtime podporu prohlížeče (např. jQuery, Bootstrap, React, mono-wasm, ...). **Zdůrazňujeme, že stejně jako v minulých letech je zakázáno používat jQuery.**

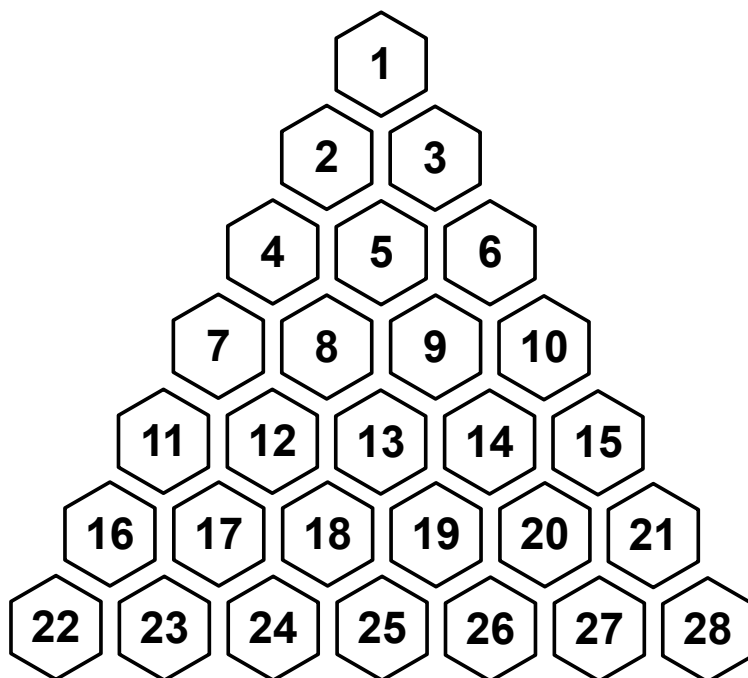
## Hexakvíz

Vytvořte webovou stránku, která umožní hrát Hexakvíz. Dobře si přečtěte celé zadání včetně dodatečných úkolů – některé dodatečné úkoly mohou ovlivnit již počáteční návrh stránky.

## Pravidla hry Hexakvíz

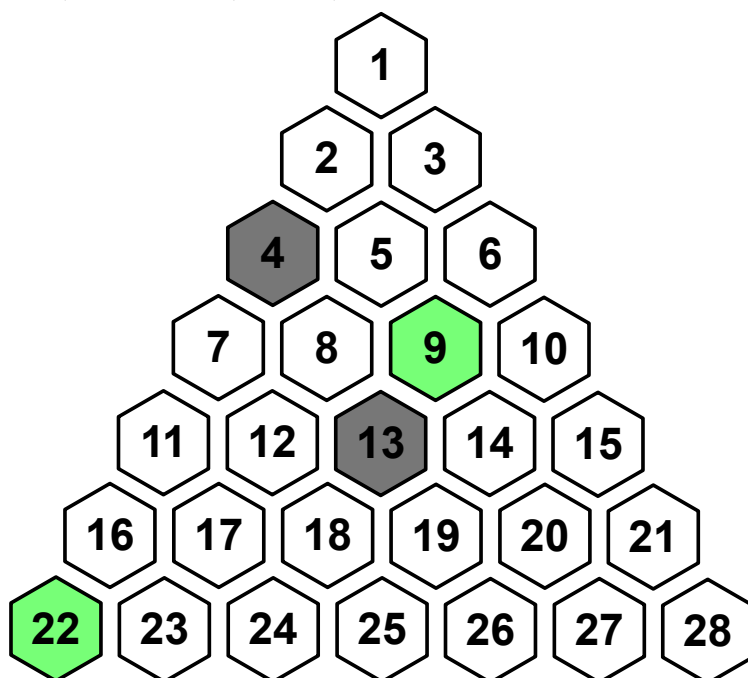
Hexakvíz je hra pro jednoho hráče. Základní hrací plocha sestává z 28 očíslovaných šestiúhelníkových políček uspořádaných do trojúhelníku, viz obrázek.

*Obr. Příklad hrací plochy*



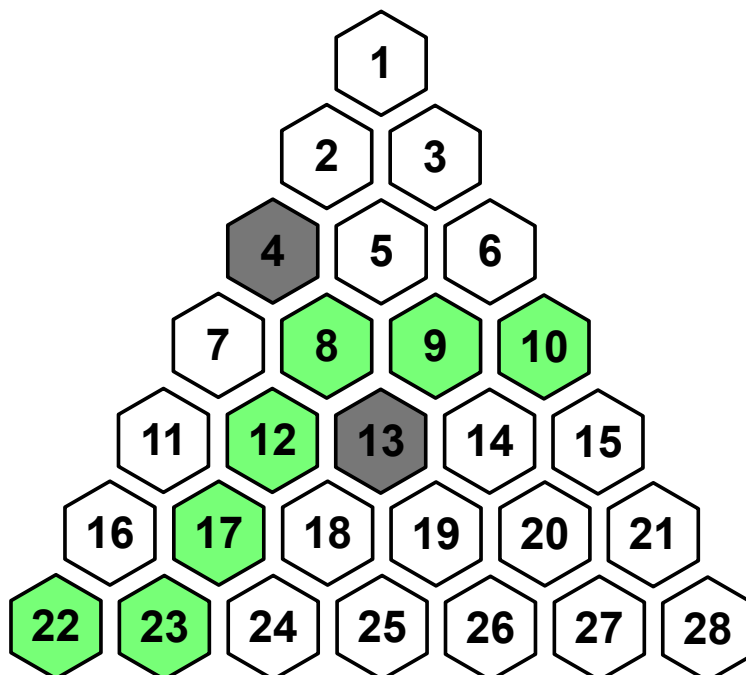
Po kliknutí na šestiúhelníkové políčko je hráči zobrazena otázka a čtyři možné odpovědi. Právě jedna odpověď je správná. Hráč má 30 sekund na to, aby si vybral jednu odpověď. Je-li správně, zbarví se příslušné políčko zeleně; je-li odpověď špatně, nebo mu to trvalo moc dlouho, zbarví se políčko šedě. Na zbarvená políčka již dále není možno kliknout.

*Obr. Rozehraný kvíz: dvě správné a dvě špatné odpovědi*

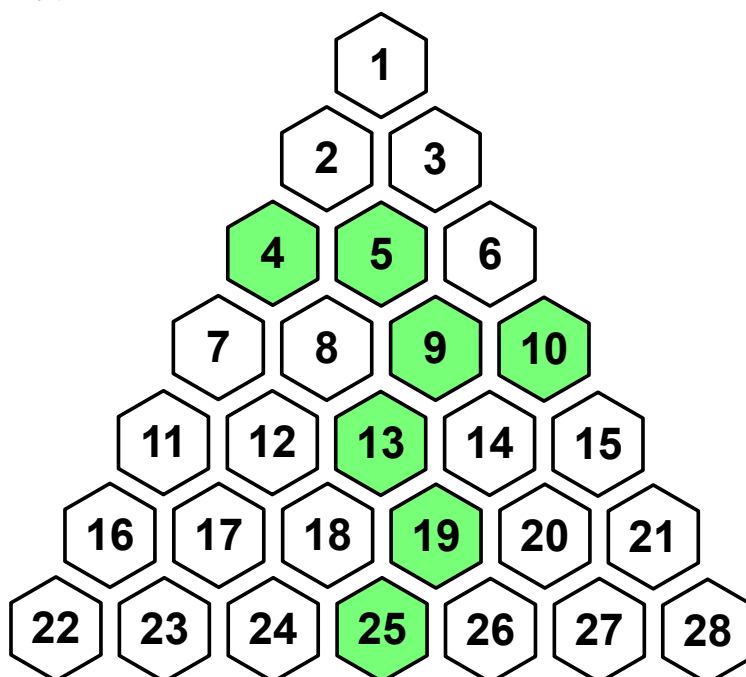


Hráč vyhraje, pokud se mu podaří propojit souvislým útvarem zelených políček všechny tři strany trojúhelníku. Rohová políčka se považují za součást obou přilehlých stran. Dva příklady vyhraného kvízu jsou zobrazeny níže.

*Obr. Vyhraný kvíz, první příklad*



*Obr. Vyhraný kvíz, druhý příklad*



### Úkoly: formátování hrací plochy

Hrací plocha je tvořena bílými šestiúhelníky s černým okrajem. Mezi sousedními šestiúhelníky je malá mezera. Hrací plocha nesmí být deformována – šestiúhelníky musejí být pravidelné a musejí mít mezi sebou stejné rozestupy.

Hrací plocha je ve středu okna prohlížeče. Má největší možnou velikost, aby se do okna vešla bez nutnosti skrolování. Stránka reaguje na změnu velikosti okna: hrací plocha se zvětšuje či zmenšuje tak, aby stále zabírala maximum okna, a stále je uprostřed.

Pokud myš přejede políčko, políčko se trochu zvětší a cinkne. Po odjetí myši se zase zmenší (a už necinká).

Po kliknutí se políčko plynule zvětší přes celé okno (animace) a místo čísla se v něm zobrazí otázka se 4 možnostmi odpovědi. Odpovědi jsou označeny písmeny A, B, C a D.

Pokud hráč klikne na správnou odpověď, tak se její text nebo pozadí plynule zabarví zeleně. Pokud hráč klikne na špatnou odpověď, zvolená odpověď se plynule zabarví červeně a zároveň (najednou) se plynule zabarví zeleně správná odpověď. Během toho je přehrán vhodný zvuk – různý pro správnou a pro špatnou odpověď.

Po vteřině se políčko opět plynule zmenší a přitom plynule mění barvu na šedou nebo zelenou.

Během odpovídání běží časomíra. Časomíru můžete zobrazit buď jako vodorovný pruh, který postupně ubývá, nebo jako postupně ubývající kružnici kolem hrací plochy. Kružnice je hodnocena více body.

Při výhře je hrací plocha ztmavena, je přes ni zobrazen nápis VÝHRA a je přehrán vhodný zvuk. Pod nápisem je tlačítko "Hrát znovu", které spustí novou hru.

## Úkoly: mechanismus hry

Otázky načítejte ze serveru ve formátu JSON. Popis API naleznete v samostatné kapitole níže. Otázku vždy načtete až v okamžiku, kdy ji potřebujete zobrazit; nenačítejte otázky dopředu. Správnost či nesprávnost odpovědi vyhodnocuje server.

Pokud nechcete načítat otázky ze serveru a spokojíte se s méně body, můžete mít otázky pevně zabudované ve stránce. Skupinu statických otázek naleznete v samostatné kapitole níže.

Na odpověď má hráč 30 sekund; poté vyhodnoťte odpověď jako špatnou. Během odpovídání je zobrazena časomíra.

Detekujte výhru (propojení tří stran) a zobrazte informaci o výhře.

Detekujte prohru, tedy nemožnost strany propojit. Zobrazte informaci o prohře obdobně jako informaci o výhře.

## Dodatečné úkoly

Za dodatečné úkoly můžete získat body navíc.

- Šestiúhelníky kolem sebe vrhají "stín", viz obrázek níže.
- Na začátku hry označte náhodně tři políčka jako špatně zodpovězená (šedě) a tři políčka jako správně zodpovězená (zeleně).
- Během hry počítejte skóre. Při správné odpovědi na otázku získá hráč za každou nevyužitou vteřinu do třicetisekundového limitu jeden bod. Příklad: hráč odpověděl na otázku správně za 17 sekund, získá tedy 13 bodů.
- Umožněte nastavení velikosti hrací plochy přes parametr `size` v URL. Velikost je nastavitelná v intervalu 4 až 13 a udává počet políček na jedné straně trojúhelníka (základní velikost hrací plochy je tedy 7). Příklad: `index.html?size=4` nastaví nejmenší možnou hrací plochu.

**Obr. Políčko vrhající stín****Popis API**

API umožňuje vytvoření hry a v rámci jedné hry potom dotazy na jednotlivé otázky a ověření správnosti odpovědí. Pro usnadnění vývoje server nezakazuje opakování dotazů (např. zopakování jedné otázky nebo vyzkoušení všech odpovědí), vaše řešení ale smí požádat o otázku nejvýše jednou, a nejvýše jednou vyhodnotit správnost odpovědi.

Nejprve je nutné založit novou hru. Toho dosáhnete odesláním požadavku metodou POST na endpoint `http://hexa-kviz.proed.cz/api/game[/size]`, kde *size* je velikost hracího pole (hranaté závorky označují volitelnou část). Odpovědí je JSON objekt s následujícími vlastnostmi:

**game**

Identifikátor hry (zakódovaný řetězec, poměrně dlouhý). Tento řetězec potřebujete pro použití ostatních endpointů, jeho obsah odesíláte v hlavičce **X-Game**.

**size**

Zvolená velikost hry (pro kontrolu).

**start**

Čas vytvoření hry.

Identifikátor hry je platný pouze půl hodiny od vytvoření.

Otázku získáte zasláním GET requestu na `http://hexa-kviz.proed.cz/api/question/policko`, kde *policko* je číslo políčka počítané od jedné. Nezapomeňte na hlavičku **X-Game**. Server vrátí objekt v JSONu s těmito položkami:

**id**

Identifikátor otázky (zakódovaný řetězec). Tento identifikátor je potřeba poslat serveru zpátky spolu s odpovědí.

**question**

Otázka (řetězec).

**answers**

Pole se čtyřmi řetězci. Každý řetězec představuje jednu možnost odpovědi.

Pro vyhodnocení odpovědi pošlete POST request na URL `http://hexa-kviz.proed.cz/api/answer/policko`. Jako tělo požadavku odešlete JSON objekt s položkami:

**id**

Identifikátor otázky, tak, jak jste ho dostali od serveru s otázkou.

**answer**

Číslo 0 až 3, jedná se o index do pole s otázkami, které jste dostali od serveru.

Pokud jste vše udělali správně, server vám jako odpověď na POST request vrátí JSON objekt s položkami:

**result**

1 (správně) nebo 0 (špatně). Pokud je 1, žádné další položky už objekt neobsahuje.

**correct\_answer**

Index správné odpovědi (pouze pokud byla odpověď špatně nebo pozdě).

reason

"timeout" nebo "wrong answer".

Pokud jste něco udělali špatně, server vám vrátí 403 Forbidden, nebo 400 Bad Request.

#### Příklad komunikace:

```
> POST /api/game/7 HTTP/1.1
> ...

< HTTP/1.1 200 OK
< ...

{"game":"a0j9nidL4zF\
\\wFq2ZSNJCtiSFES2ROv2AwNaqXaZRG0pXBd5a9COHx1WARZZ8oImrIbqLeXZ7EvXywwKLyQ=",
"size":7, "start":"2021-06-19T09:00:00.000000"}
```

```
> GET /api/question/1 HTTP/1.1
> X-Game: a0j9nidL4zF
\\wFq2ZSNJCtiSFES2ROv2AwNaqXaZRG0pXBd5a9COHx1WARZZ8oImrIbqLeXZ7EvXywwKLyQ=
> ...

< HTTP/1.1 200 OK
< ...

{"answers":["Správně", "Špatně", "Špatně", "Špatně"],
"id":"ocpFL2vuvXInzwPH6BDSGcx11y
\\\"jp9MdGxQPN96YpdWyachgIR5FQ8JOa2w05TAXWRUZBuR0zJ4PB1M=", "question":"Co
je správná odpověď?"}
```

```
> POST /api/answer/1 HTTP/1.1
> X-Game: a0j9nidL4zF
\\wFq2ZSNJCtiSFES2ROv2AwNaqXaZRG0pXBd5a9COHx1WARZZ8oImrIbqLeXZ7EvXywwKLyQ=
> Content-Type: application/json
> ...

{"id":"ocpFL2vuvXInzwPH6BDSGcx11y
\\\"jp9MdGxQPN96YpdWyachgIR5FQ8JOa2w05TAXWRUZBuR0zJ4PB1M=", "answer":1}

< HTTP/1.1 200 OK
< ...

{"result":0, "correct_answer":0, "reason":"wrong answer"}
```

#### Statické otázky

Statické otázky slouží jako záchrana. Za řešení se statickými otázkami bude přiděleno méně bodů než za řešení využívající API. Pole s definicí otázek lze stáhnout na <http://hexa-kviz.proed.cz/api/fallback/cislo>, jako číslo použijte např. své soutěžní.