Soutěž dětí a mládeže v programování

Obvodní kolo Prahy 1 a 2, rok 2026

Autoři: Adam Benda, Martin Horský, David Šertler; verze 1.0, 2026-02-05

Každá úloha má koeficient obtížnosti, kterým se násobí hodnocení vašeho řešení. Odevzdejte nám prosím řešení přes odevzdávátko (https://soutez.lnker.xyz/) jako komprimovanou složku zip. Odevzdat jednu úlohu můžete vícekrát - nejlépe po dosažení nějakého stupně funkčnosti. Hodnotíme verzi úlohy, kterou jste odevzdali jako poslední. Kromě splnění zadané funkčnosti se hodnotí přehlednost vašeho kódu (komentáře, popisující názvy proměnných) a efektivita vašeho řešení.

Při soutěži je zakázána komunikace (přítel na telefonu). Vyhledávač a Internetové zdroje používat můžete, stejně jako generátory kódu. Zkopírovat si několik řádků kódu ze stackoverflow (uveďte zdroj) nebo použít knihovní funkci řešící dílčí část problému je v pořádku. V případě pochybností konzultujte porotce.

Používání knihoven je povoleno, jelikož mnoho programovacích jazyků má relativně omezenou standardní knihovnu (např. grafické zobrazování v okýnkách). Nepoužívejte ale knihovny, které víceméně vyřeší úlohu za vás.

Smíte používat LLM chatboty. Pracujte s nimi ale opatrně. Obecně je dobré se nespoléhat vždy na výstup LLM. Tyto AI můžou vkládat do svého výstupu chyby (AI halucinace). Měli byste z nich používat pouze kód, který sami chápete.

Testovací vstupy k úlohám naleznete na https://soutez.lnker.xyz/res/.

Po souteži nám prosím vyplňte anketu na https://lnker.xyz/anketa.

Úloha 1: Slovní fotbal - hra (koef. 2)

Eda miluje slovní fotbal. Je sečtělý a má neuvěřitelnou slovní zásobu. Bohužel s ním kvůli tomu už nikdo nechce hrát. Zkuste to s ním: AKORD - DÁMA - ANANAS - SLUCH - CHOBOT - …

Připravte Edovi hru s počítačovým protihráčem.

- Počítač používá slova ze slovníku (txt/json). Slovo musí začít posledním písmenem slova předcházejícího
- Člověk může použít libovolné slovo začínající posledním písmenem slova předcházejícího. Pokud neví, může hru vzdát. Není potřeba slovo kontrolovat vůči slovníku (předpokládá se, že člověk hraje čestně a že ve slovníku nemusí být vše)
- Ani jeden ale nesmí ve hře zopakovat slovo které již bylo uvedeno
- Nezáleží na velikosti písmen, např. “Slovo”, “slovo” a “SLOVO” jsou totéž

- Program si musí poradit s diakritikou a českými slovy. Slovník používá kódování UTF-8
- “Ch” je jedno písmenko

Hodnocení:

- Interaktivní smyčka 10%
- Počítač správně používá slova ze slovníku 40%
- Detekce prohry a mezní případy 40%
- Přehlednost a efektivita kódu 10%

Úloha 2: Slovní fotbal - nejdelší řetěz (koef. 3)

Leonard taky miluje slovní fotbal.

Lze ze zadaného slovníku odehrát partii slovního fotbalu, která použije všechna slova? Rozhodněte (ano/ne) a v případě kladné odpovědi vypište řetěz slov.

V této úloze už nemusí hrát dva hráči. Používáte pouze slova ze zadaného formátu (stejný formát jako v předchozí úloze).

Slovník může mít i milion slov. Hodnotí se i paměťové a výpočetní nároky vašeho programu - tedy za jak dlouhou dobu pro tak velký vstup doběhne.

Hodnocení:

- Správné řešení (ano/ne) 30%
- Vypsání slov pro splnitelné zadání 30%
- Efektivita řešení 20%
- Přehlednost a elegance kódu 20%

Úloha 3: Defragmentace (koef. 3)

Zastaralé systémy státní správy jsou peklo. Ale dokud fungují tak je nikdo nechce vyměnit. Tady je chtějí vyměnit. Pomůžete naším úředníkům přejít na nový software?

Narazili jste na prapodivný počítačový systém. Vypadá to že při vytváření souborů se v něm musí už na začátku nastavit velikost a nejde zvětšit. Občas, ale potřebujete soubory zvětšovat. Sekretářky, které systém používají se naučily soubory fragmentovat. Teď se ale se systémem pracuje pomalu a je potřeba obsah systému defragmentovat.

Vaším vstupem je cesta k souboru a stromová struktura souborů. Všechny použité soubory jsou textové. Mohou obsahovat libovolný text, ale speciální význam mají hranaté závorky. Všechny hranaté závorky mají být zaměněny za jiný text. V hranatých závorkách je cesta k souboru, jehož obsah má zaměnit hranaté závorky. Tento vložený soubor může ale také obsahovat hranaté závorky, které vyjadřují odkaz do jiného souboru.

Vaším cílem je určit celý obsah souboru včetně záměn. Pokud máme soubory A.txt, B.txt a C.txt s následujícím obsahem:

A.txt = “`Lorem ipsum dolor [B.txt] sit amet [C.txt].`”

B.txt = “`Josef [C.txt]`”

C.txt = “`Vyskočil`”

A musíme defragmentovat soubor A.txt, tak výsledek by měl být:

“`Lorem ipsum dolor Josef Vyskočil sit amet Vyskočil.`”

Soubory nemusí být všechny v jedné složce. Struktura souborů může být v mnoha složkách. Odkaz do jiného souboru je vždy relativní k aktuálně zpracovávanému souboru.

Jelikož mohou na sebe tyto soubory odkazovat libovolně, může se stát že některé soubory na sebe odkazovat ve smyčce. V takovém případě by měl váš program skončit a nahlásit chybu.

Cestu ke vstupnímu souboru váš program může brát na příkazové řádce nebo skrz grafické rozhraní a výstup může vracet na standardní výstup, do souboru nebo v grafickém rozhraní. Toto rozhodnutí je na vás, ale uveďte je v dokumentaci.

Můžete předpokládat že celková velikost souborů a celková velikost výsledku dohromady nepřekročí 500MiB, počet souborů nepřekročí 5000 a hloubka odkazů nepřekročí 100.

Hodnocení:

- Správná defragmentace v jedné složce - 30%
- Správné defragmentace v mnoha složkách - 20%
- Detekce smyčky - 20%
- Detailní výpis smyčky - 20%
- Efektivita a dokumentace - 10%

Úloha 4: Analogové hodiny (koef. 3)

Někdo sleduje jen jedny hodiny na zdi, jiný ale potřebuje vědět, kolik je hodin na více místech světa najednou. Někdo má rodinu v zahraničí, někdo pracuje s kolegy v jiných časových pásmech a někdo si prostě jen chce hlídat, kdy je správný čas zavolat kamarádovi do ciziny. Obyčejné hodiny na to už dnes nestačí.

Vaším úkolem je vytvořit program, který zobrazuje ručičkové (analogové) hodiny s ciferníkem. Hodiny musí obsahovat kruh ciferníku, značky nebo čísla 1 až 12 na odpovídajících pozicích a tři ručičky (hodinovou, minutovou a sekundovou). Tyto hodiny musí ukazovat čas a jejich ručičky se musí v čase správně a průběžně pohybovat. Program po spuštění zobrazí alespoň jedny takové hodiny s aktuálním časem.

Program ale musí umožnit pracovat s více hodinami najednou. Uživatel si může vytvořit libovolný počet hodin, přičemž každé z nich mají své jméno (například „Praha“, „New York“, „Tokyo“, „Chata“). Každé hodiny mohou ukazovat jiný čas - buď pomocí posunu oproti místnímu času (časové pásmo), nebo mohou mít ručně nastavený vlastní čas, který je na ostatních hodinách nezávislý. Uživatel musí mít možnost hodiny přidávat, odebírat a upravovat jejich název i nastavení času. Všechny vytvořené hodiny musí běžet současně a jejich zobrazení se musí průběžně aktualizovat.

Hodnocení:

- Správné zobrazení a běh analogových hodin (ciferník, čísla/značky, pohyb ručiček) - 50%
- Správná práce s více hodinami současně (přidání/odebrání, jména, nezávislé nastavení) - 40%
- Efektivita a dokumentace - 10%