

Zadání okresního kola okresu Uherské Hradiště v Soutěži v programování – kategorie mládež i žáci 2015

Na svém disku vytvořte složku – Příjmení2015 a podsložky jednotlivých úloh, v závěru uložíte tuto složku na společný disk nebo flaskdisk.

1. Hádání čísla

Dva hráči – Hráč A a Hráč B spolu soupeří, kdo dříve uhodne číslo, získané z hodu kostkami. Při hodu máme k dispozici 6 kostek. Při prvním hodu hodíme pouze jednou z kostek a číslo, které padne(1-6), určí, kolikrát budeme následně házet zbývajících 5 kostkami. Při těchto hodech pak vždy sečteme všechna čísla, která padnou na 5 vržených kostkách a výsledné součty pak průběžně sčítáme až do provedení všech požadovaných hodů. Tím získáme číslo, které budou následně hráči hádat.

Příklad:

Hodíme jednou kostkou a padne číslo 3.

Proto následně házíme zbylými pěti kostkami třikrát a padnou čísla:

5 1 6 1 3 (součet 16)

4 5 4 2 3 (součet 18)

3 3 5 1 3 (součet 15)

Výsledným číslem, které budou hráči hádat, se tak stává číslo 49.

Program nejprve provede náhodný „hod“ kostkami a vypočítá hádané číslo, poté hra začne.

Oba hráči se střídají v tipování čísla (na začátku program provede vylosování, který hráč bude začínat) do té doby, než jeden z hráčů číslo uhodne. Po tipnutí čísla hráčem dá program vždy jednu ze 3 možných odpovědí:

- hádané číslo je menší,
- hádané číslo je větší,
- trefa – uhodls.

Po uhodnutí čísla jedním z hráčů program vypíše, jak hádané číslo vzniklo (ukáže, jaká čísla byla na kostkách při všech provedených hodech).

Vítěz kola si připsá bod a hra se opakuje od začátku – program vygeneruje nové číslo a hráči znovu tipují (začínají přitom v opačném pořadí než v předchozím kole). Vítězem se stává hráč, který první získá 2 body.

Po skončení celé hry program vypíše:

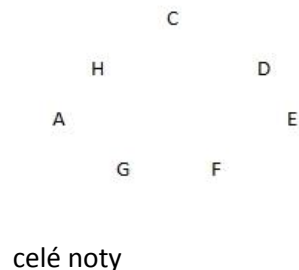
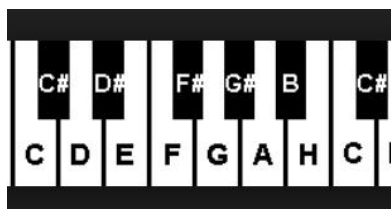
- výsledné skóre (průběžné skóre by se mělo objevovat na obrazovce po celou dobu hry)
- posloupnosti čísel, představující tipy obou hráčů v jednotlivých kolech až do uhodnutí čísla
- které kolo bylo nejrychlejší (při kterém kole byl potřeba k uhodnutí čísla nejmenší počet pokusů).

2. Stupnice

Hugo hraje na klavír, má výbornou paměť a potřebuje se naučit všechny stupnice. My mu pomůžeme a naprogramujeme všechny stupnice, on si je bude postupně nebo náhodně zobrazovat jako názvy jednotlivých not a pomocí notového zápisu.

Princip durových stupnic s křížky a s béčky:

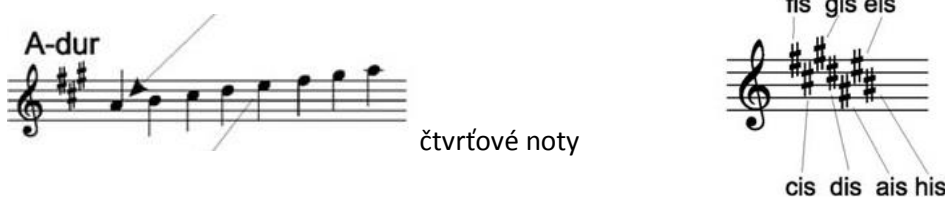
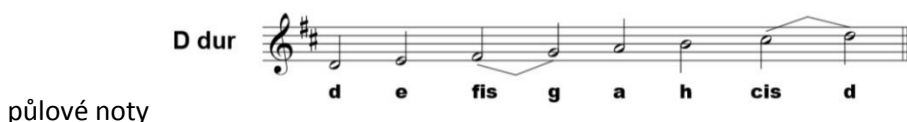
Durová stupnice je dána 8 tóny a mezi 3. a 4. je půltón a také mezi 7. a 8. (C dur mezi E F a H C) Doporučuji pro pochopení zapsat noty do kruhu a postupně je přepisovat a tím vznikají nové stupnice.



A) Další nová stupnice s křížky začíná od kvinty (s jedním # od G – to je stupnice G dur) a když potřebujeme celý tón tak ho zvýšíme #, např. F na Fis.

G dur: G A H C D E Fis G (půl tóny jsou mezi H C a Fis G)

Další stupnicí se 2 křížky je D dur



B) Další nová stupnice s béčky začíná od kvarty (s jedním b od F – to je stupnice F dur) a když potřebujeme půltón, tak ho snížíme b, např. H na Hes, ten se ale nazývá B.

F dur: F G A B C D E F (půltóny jsou mezi A B a E F)

Další stupnicí se 2 b je B dur



Máme tyto úkoly:

1. Vypište a zobrazte notový zápis stupnice C dur:
 - a) názvy not C D E F G A H C
 - b) notovým zápisem (nakreslete 5 linek, houslový klíč a noty)
2. Postupně generujte všechny durové s 1 – 7 křížky nebo béčky po zadání pokynu „další“, jednotlivé křížky ‘#’ či ‘b’ zakreslujte za houslovým klíčem, ten můžete použít hotový.
3. Zobrazte zadanou stupnici na vstupu, durovou stupnici C dur nebo s 1 – 7 křížky nebo 1-7 béčky.
4. Náhodně zobrazujte libovolnou durovou stupnici C dur nebo s 1 – 7 křížky nebo 1-7 béčky, název stupnice až po volbě – kontrola
5. Vytvořte výstupní soubor, který bude všechny stupnice přehledně znázorňovat.
6. Pro zpestření zakreslujte po výběru noty celé, půlové nebo čtvrté

3. Úprava souboru XML

Cermat potřebuje pomoci s přijímačkami. Data, která byla exportována z Bakalářů je potřeba trochu upravit. Vstupní soubory vstup4.xml a vstup8.xml upravte na vystup4.xml a vystup8.xml. Pro hodnocení budou použity jiné soubory.

Extensible Markup Language (zkráceně **XML**, česky *rozšiřitelný značkovací jazyk*) je obecný značkovací jazyk, který byl vyvinut a standardizován konsorciem W3C. Je zjednodušenou podobou staršího jazyka SGML. Umožňuje snadné vytváření konkrétních značkovacích jazyků (tzv. aplikací) pro různé účely a různé typy dat. Používá se pro serializaci dat. Zpracování XML je podporováno řadou nástrojů a programovacích jazyků.

Jazyk je určen především pro výměnu dat mezi aplikacemi a pro publikování dokumentů, u kterých popisuje strukturu z hlediska věcného obsahu jednotlivých částí, nezabývá se vzhledem.

Úkol spočívá v tom, že v textovém souboru na určitých pozicích vyměňujete data.

Zde je část XML Souboru:

Zpracujte tyto úkoly:

1. K evidenčnímu číslu všem přidat 800 nebo 400, podle toho, zda nastupují do osmiletého typu studia nebo čtyřletého, typ studia vložte nebo přečtěte z názvu vstupního souboru.
2. Obor studia u všech definujte 79-41-K/41 pro čtyřletý typ a 79-41-K/81 pro osmiletý typ. Nahradte obor G41 a G81
3. Ročník, z kterého nastupují, definujte všem 5 nebo 9, podle typu studia (5 pro osmiletý, 9 pro čtyřletý)
4. V bydlišti zaměňte čárku mezi obcí a ulicí za středník a číslo domu oddělte od ulice také středníkem. Tyto opravy přehledně zobrazte na obrazovce (původní -> nový)
5. IZO školy upravte na text, a pokud neobsahuje 9 cifer, tak je zepředu doplňte nulami. Např. 12345678 změňte na '012345678'