

Hodnocení soutěžních úloh

Kategorie programování žáci a mládež

20. až 22. dubna 2017

Soutěž v programování – 31. ročník

Krajské kolo 2016/2017

[Prasátko](#)

[Dvanáct set troska](#)

[Simulátor Enigma](#)

Prasátko

Koeficient 1

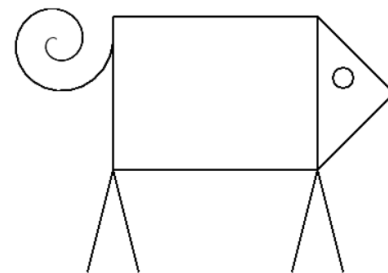
Prohlédněte si prosím animaci v přiloženém souboru [prase.html](#). Vaším úkolem je napsat program, který zobrazí obdobný výsledek. Nejvíce bodů získáte, pokud provedete plynulou animaci vykreslování čar (za stejný čas se vždy vykreslí stejná délka čáry). Pozornost věnujte i postupnému vykreslení oka a ocásku, u nich se hodnotí kulatost jak oka, tak ocásku a u ocásku také shoda s předlohou (správný směr zakroucení, stoupání spirály, počet otáček).

Pokud je celé zadání nad vaše síly, můžete zpracovat jednodušší variantu – animovat postupně po automaticky spouštěných krocích, kdy krokem je zobrazení jedné celé čáry, a pak celého oka a celého ocásku. Několik málo bodů dostanete i v případě, že program celé prasátko nakreslí najednou.

Body dostanete i za to, že v průběhu animace okno vašeho programu „nevytuhne“ a lze s ním např. pohybovat.

Je zakázáno využívat zdrojový kód vzorového řešení a obrázek prasátka ze zadání. Nebudou také hodnoceny programy zobrazující předpřipravené obrázkové soubory.

Pokud budete mít problém se zobrazením vzorové animace, můžete vyjít z přiloženého obrázku a následujícího popisu: Animace začíná vykreslením čáry za hlavou prasátka shora dolů, následuje spodní hrana těla, zadní hrana a pak hrana horní. Následně dvě čáry hlavy od té horní (začíná shora), přední nohy, zadní nohy, oko a nakonec ocásek. Nohy a ocásek se vždy začínají kreslit na straně blíže k tělu. Pokud na sebe dvě čáry navazují (tělo, hlava), začíná se ta druhá kreslit ve směru od té první.



Hodnocení

Pokud jsou použity předpřipravené obrázkové soubory, tak úlohu nehodnotíme (získá 0 bodů).

Body	Za co
0.5	zobrazí obrázek, minimálně tělo a hlavu
0.5	zobrazí oko
0.5	zobrazí ocásek
1.0	oko není zubaté (je vykreslené jako kružnice)
1.0	ocásek není zubatý (viditelně tvořený z čar)
1.0	ocásek odpovídá předloze
1.0	existuje animace (po krocích nebo plynulá)
2.5	animace je plynulá (při viditelném zpomalené/zrychlení animace u kratších prvků ubrat jeden bod)
1.0	během animace okno nevytuhne (a pokud animace vůbec není, bod také neudělujeme)
1.0	dokumentace a přehlednost zdrojového kódu

Dvanáct set troska

Koeficient 3

Zdědili jste po svém pradědečkovi luxusní automobil Škoda 1203. Poté, co jste spatřili své dědictví, které jste ušlechtilé nazvali *dvanáct set troska*, rozhodli jste se ho odvézt zhodnotit na skládku. Po cestě se stala ale nemilá věc – zasekl se vám volant a nejde otočit doleva, můžete tedy na křižovatkách jezdit jen rovně nebo zatáčet doprava! Vaším úkolem je s takto poškozeným vozem najít co nejkratší cestu na skládku, aby se vám náhodou po cestě nerozbilo ještě něco.

Napište program, který na vstupu dostane mapu města, výchozí pozici dvanáct set trosky a pozici skládky, a najde nejkratší cestu dvanáct set troskou na skládku tak, aby na každé křižovatce jela buď rovně nebo zahla doprava. Vypište délku nejkratší cesty a *interaktivně* nalezenou cestu zobrazte.

Vstup

Vstup načtete ze souboru `mesto.txt`. Tento soubor obsahuje mapu města, což je čtverečková síť o rozměrech 100×100 políček. Mapa města je uložena na 100 řádcích s tím, že každý řádek je ukončený znaky CR a LF a obsahuje (kromě znaků ukončení řádku) přesně 100 znaků – jednotlivá políčka tohoto řádku mapy města. Znaky v souboru mohou být následovné:

- . silnice (průjezdné pole)
- x budova (neprůjezdné pole)
- @ skládka
- % dvanáct set troska

Můžete předpokládat, že soubor je vždy v pořádku, obsahuje právě jednu skládku a právě jednu dvanáct set trosku. Navíc počáteční pozice dvanáct set trosky sousedí s právě jednou silnicí – předpokládejte tedy, že dvanáct set troska je na začátku natočena směrem k této silnici.

Výstup

Pro daný soubor najděte nejkratší cestu dvanáct set trosky na skládku tak, aby dvanáct set troska jela jen po silnici a zatáčela jen doprava (tj. pokud z nějakého směru přijede na políčko silnice, může buď pokračovat rovně, nebo

zatočit o 90° doprava). Můžete předpokládat, že taková cesta vždy existuje. Pokud jich existuje více, najděte libovolnou z nich.

Až cestu naleznete, vypište její délku – počet políček, které na cestě navštívíte, včetně políčka skládky. Pokud nějaké políčko navštívíte vícekrát, počítejte ho vícekrát. Poté nalezenou cestu interaktivně zobrazte – umožněte jednak vykreslení kompletní cesty, a jednak krokování jednotlivých políček cesty.

Příklad

Pro následující mapu města (pro názornost je zobrazený jen její levý horní roh o velikosti 7×7, všechna ostatní nezobrazená políčka jsou x):

```

XXXXXXXX
.....
.X.X.X.
.....
.X.X.X.
.X@X%X.
XXXXXXXX

```

je nejkratší cesta na skládku dlouhá 22 políček. Kdyby bylo možné zatáčet doleva, byla by nejkratší cesta dlouhá 6 políček. Nejkratší správná cesta je zobrazena na následující mapě:

```

XXXXXXXX
|X|X|X|
|X|X|X|
.X|X|X.
.X@X%X.
XXXXXXXX

```

Hodnocení

Program soutěžících se vyhodnotí na devíti vstupních souborech mesto1.txt až mesto9.txt, za každý dostane soutěžící 0 až 1 bod.

Při vyhodnocování souboru mestoN.txt zkopírujte tento soubor do adresáře s programem soutěžícího a přejmenujte ho na mesto.txt. Program soutěžícího spusťte a pokud to program vyžaduje, vyberte soubor mesto.txt. Program musí nalézt nejkratší cestu do pěti vteřin. Pokud program v limitu vypsál správnou

délku nejkratší cesty, dostane 0.5 bodu. V případě vypsání správné délky nejkratší cesty dostane dalších až 0.5 bodu za interaktivní zobrazení nejkratší cesty dle hodnocení poroty, s tím, že plný počet bodů by mělo dostat řešení, které zobrazuje jak kompletní cestu, tak umožňuje pohodlné krokování políček cesty tam i zpět. Pokud je zobrazená cesta zjevně nekorektní, řešení dostane za interaktivní zobrazení 0 bodů.

Body	Za co
0.5	Program v limitu vypíše délku nejkratší cesty 50 pro soubor mesto1.txt
0.5	Program v limitu vypíše délku nejkratší cesty 68 pro soubor mesto2.txt
0.5	Program v limitu vypíše délku nejkratší cesty 1274 pro soubor mesto3.txt
0.5	Program v limitu vypíše délku nejkratší cesty 206 pro soubor mesto4.txt
0.5	Program v limitu vypíše délku nejkratší cesty 292 pro soubor mesto5.txt
0.5	Program v limitu vypíše délku nejkratší cesty 5049 pro soubor mesto6.txt
0.5	Program v limitu vypíše délku nejkratší cesty 516 pro soubor mesto7.txt
0.5	Program v limitu vypíše délku nejkratší cesty 680 pro soubor mesto8.txt
0.5	Program v limitu vypíše délku nejkratší cesty 670 pro soubor mesto9.txt
až 0.5	Program interaktivně zobrazí nejkratší cestu pro soubor mesto1.txt
až 0.5	Program interaktivně zobrazí nejkratší cestu pro soubor mesto2.txt
až 0.5	Program interaktivně zobrazí nejkratší cestu pro soubor mesto3.txt
až 0.5	Program interaktivně zobrazí nejkratší cestu pro soubor mesto4.txt

Body	Za co
až 0.5	Program interaktivně zobrazí nejkratší cestu pro soubor mesto5.txt
až 0.5	Program interaktivně zobrazí nejkratší cestu pro soubor mesto6.txt
až 0.5	Program interaktivně zobrazí nejkratší cestu pro soubor mesto7.txt
až 0.5	Program interaktivně zobrazí nejkratší cestu pro soubor mesto8.txt
až 0.5	Program interaktivně zobrazí nejkratší cestu pro soubor mesto9.txt
až 1	dokumentace, komentáře, přehlednost, výstižné názvy proměnných, ...

Simulátor Enigma

Koeficient 2

Enigma byl šifrovací přístroj, používaný za druhé světové války. Tento šifrovací stroj uměl šifrovat vstupní text pomocí šifrování jednotlivých znaků vstupního textu na jiné znaky výstupního textu.

Celkový proces šifrování strojem Enigma spočíval v postupném šifrování jednoho znaku průchodem přes několik šifrovacích tabulek, což mohlo vypadat např. následovně:

```

E → B → I → S → M → K → J → T
N → P → U → J → R → Q → R → X
I → E → J → B → O → X → L → W
G → C → P → D → Z → G → M → Y
M → F → Y → P → G → D → X → S
A → G → L → W → C → E → K → V

```

Celkově ENIGMA → TXWYSV

První sloupec → poslední sloupec

Vaším úkolem je napsat program, který bude simulovat činnost šifrovacího stroje Enigma.

Šifrují se pouze velká písmena anglické abecedy.

Máme několik šifrovacích tabulek ve tvaru 26 dvojic velkých písmen anglické abecedy A až Z oddělených čárkou („bílé“ znaky ignorujte). Každé písmeno abecedy je v tabulce obsaženo právě dvakrát. Jednou je ve dvojici vlevo, jednou vpravo. Například:

EK, JF, TE, DA, BX, LO, AP, CJ, KG, SL, RI, UZ, QD,
OB, IV, XQ, GN, NT, HU, PR, YW, MC, VM, ZY, WS, FH

Při šifrování se znak A změní na P, B na X atd., až Z na Y.

Vstupem programu jsou textové soubory, jednotlivé řádky jsou odděleny znaky CR LF, vstupní kódování je ASCII. Výstupem programu je výpis zašifrovaného textu na obrazovku.

Postupně v programu naimplementujte následující funkce:

- Uživatel si vybere vstupní soubor pro šifrování.
- Načtení vstupního souboru a nastavení šifrování pro tento běh programu z prvního řádku.
- Vypsání nastavení šifrování.
- Vypisování nezašifrovaného a zašifrovaného textu ze vstupního souboru po jednotlivých řádkách.

První řádka vstupního souboru obsahuje jména použitých šifrovacích tabulek. Jména jsou oddělena čárkou, bílé znaky se ignorují.

Například při nastavení ve tvaru

I.A, II.A, III.A, P1.AB, III.B, II.B, I.B

budou pro šifrování postupně použity šifrovací tabulky ze souborů I.A, II.A, ... a jako poslední ze souboru I.B.

Další řádky vstupního souboru obsahují text zprávy určený k zašifrování psaný velkými písmeny anglické abecedy. Mezery se přenáší beze změny.

Po načtení nastavení šifrování program toto nastavení (obsah první řádky vstupního souboru) vypíše. Následovně bude vstupní soubor po řádkách šifrovat. Pro každou řádku vstupního souboru program vypíše dvě řádky – první

řádka bude obsahovat originální text a druhá zašifrovaný text. Nezašifrovaný řádek začíná textem VSTUP= a zašifrovaný textem VYSTUP=.

V ukázkovém adresáři máte kromě šifrovacích tabulek i několik vzorových datových souborů VZORx.DAT.

Například pro soubor VZOR6.DAT:

```
I.A, UKWB.A, I.B
SKAKAL PES PRES OVES
IMNMNW RHI RPHI FJHI
RPHI YHWHAFB WFBMB
```

by měl výstup programu vypadat takto:

```
I.A, UKWB.A, I.B
VSTUP=SKAKAL PES PRES OVES
VYSTUP=IMNMNW RHI RPHI FJHI
VSTUP=IMNMNW RHI RPHI FJHI
VYSTUP=SKAKAL PES PRES OVES
VSTUP=RPHI YHWHAFB WFBMB
VYSTUP=PRES ZELENOU LOUKU
```

Hodnocení

Body	Za co
0.5	Uživatel si může vybrat vstupní soubor
0.5	Zobrazí se první řádek vstupního souboru
1	zpracuje soubor POROTA1.DAT (pes jitnicku sezral,...
2	zpracuje soubor POROTA1.DAT a zpracuje správně mezery - vsechny tři řádky stejně dlouhé
1	zpracuje soubor POROTA2.DAT vstup=vystup = ABECEDA A,B,...,Z
1	zpracuje soubor POROTA3.DAT THE QUICK BROWN FOX JUMPS OVER A LAZY DOG
1	zpracuje soubor POROTA4.DAT NEZAPOMEN NA DOKUMENTACI ZA TO JSOU TAKY BODY ABCDEFGHIJKLMNOPQRSTUVWXYZ

Body	Za co
2	zpracuje soubor POROTA5.DAT O NAHLY DEST JIZ ZVIRIL PRACH A CILA LAN TED BEZI S HOUFCEM GAZEL K UKRYTUM
1	dokumentace, komentáře, přehlednost, výstižné názvy proměnných, ...