



## Korešpondenčný seminár z programovania XXIV. ročník, 2006/07

Katedra základov a vyučovania informatiky FMFI UK,  
Mlynská Dolina, 842 48 Bratislava

KSP finančne podporujú: aSc – Applied Software Consultants spol. s r.o.

MICROSTEP-MIS spol. s r.o.

Gnome spol. s r.o.

Whitestein Technologies spol. s r.o.

### Príklady 1. kola zimnej časti

Milé naše riešiteľky, milí naši riešitelia.

Určite vám celé leto chýbali zadania KSP a tak s príchodom nového školského roka vám prinášame nové zadania. A už sú tu. Lepšie; lepšie zatriedené: ku kategóriam Z a O sa teraz pridala aj úplne nová: kategória T. Preto si dobre prečítajte **pravidlá** na začiatku letáku.

Pevne veríme, že sa pustíte do riešenia úloh, ktoré sme pre vás pripravili. Nezabúdajte, že najlepších čaká pozvánka na sústredko, a tých ešte lepších čakajú aj nejaké tie ceny.

**Termín** odoslania riešení tejto série je pondelok **23. októbra 2006**. Riešitelia, ktorí sa za výsledky v minulom polroku dostali na jesenné sústredko, môžu riešenia odovzdať **pri príchode na sústredko**.

A ešte nepovinná domáca úloha: Ak chcete, napíšte nám na samostatnom liste papiera, čo si myslíte, že predstavujú písmená Z, O a T v názvoch kategórií.

Ako sa do hory volá, tak sa to v nej ozýva.

KSPáci

#### 1. Zaveštíme si trošičku...

kat. Z; 15 bodov

Karol šiel večer domov. Bolo už celkom neskoro. Tma ako v rohu. Čudesné zvuky. Tiene. Keď tu zrazu vidí malé svetielko. Také naozaj maličké. Ide k nemu bližšie. Postupne začína rozoznávať jemné obrysy okienka. Došiel k nemu. Otvoril dvere a vošiel dnu. To, čo tam videl, ho šokovalo. Videl svoju budúcnosť. Iba za 99,90.

Chcel vedieť viac. Tak začal googliť, až sa nakoniec dogooglil.<sup>1</sup> Celé si to prečítal a vyskúšal. A naozaj, vyšlo mu presne to, čo mu tam vyveštili. To je ono! To celý čas hľadal! Zobral dátum narodenia svojej sestry, a tá hneď v jeho očiach stúpila, keď mu vyšlo, že je inteligentná a ťažko pracujúca.<sup>2</sup> A tak to pokračovalo ďalej a ďalej.

Napokon sa Karol rozhodol, že si z toho urobí biznis. Urobí webstránku, ktorá bude ešte lepšia ako tá, ktorú našiel. Návštevníkovi jeho *magickú význačnú číslicu* rovno vypočíta.

Zišlo by sa ešte objasniť, ako sa Karolova *magická význačná číslica* počíta. Je to veľmi jednoduché. Zoberiete dátum narodenia človeka<sup>3</sup> a sčítate všetky jeho číslice. Ak má výsledok viac ako jednu číslicu, tento postup s ním znovu a znovu zopakujete, až kým nedostanete jednu jedinú číslicu.

Ak sa napríklad Karol narodil 31. 2. 1983, sčítaním číslic dostaneme výsledok 27. S číslom 27 postup zopakujeme a dostaneme, že Karolova *magická význačná číslica* je  $2 + 7 = 9$ .

**Úloha:** Napíšte program, ktorý načíta číslo  $N$  a vypočíta jemu zodpovedajúcu *magickú význačnú číslicu*.

<sup>1</sup><http://www.visionsofheaven.com/articles.docs/ARnumbers.html>

<sup>2</sup>A to ešte neskúšal ani sčítať čísla písmen v jej mene...

<sup>3</sup>Alebo jeho telefónne číslo, veľkosť topánky, či čokoľvek iné, čo zadá do okienka.

Tip: Postup spomenutý v zadání nie je najrýchlejší možný postup.

**Príklad:**

**Vstup:**

$N = 3121983$

**Výstup:**

9

## 2. Zničený autobus

kat. Z; 10 bodov

Vrrrrr vrr vrr vrrzg vrr, čik pssss dup dup dup... Dovolíte? Dup dup dup, crrrrn psss vrrr... Realita MHD, keď všetko funguje tak, ako má. „Prásk!“ Toto už v pláne nebolo. Vlna rozčúlenia medzi cestujúcimi rastie a ujo vodič sa len bezmocne máva rukami.

„Večera!“, ozve sa zrazu z vedľajšej izby a Jožko, silný to priaznivec Transport Tycoon, je nútený odpútať oči od monitora a dať si zopár chlebov. Jožko ale zároveň zároveň raz za čas skúša aj programovať, a medzi druhým a tretím kúskom chleba sa mu v hlave zrodila myšlienka – skúsiť si naprogramovať vlastnú simuláciu dopravy. Ale čo najskutočnejšiu!

**Úloha:** Na vstupe je celé číslo z intervalu  $(0, 100)$ , ktoré reprezentuje spoľahlivosť prostriedku MHD. Po jeho zadání bude po obrazovke sám jazdiť autobus, ktorý sa bude na základe uvedenej spoľahlivosti kaziť, ako by sa kazil skutočný autobus. (Hint: autobus kaziaci sa pravidelne každých 15 sekúnd sa ako skutočný nespráva :-)).

Pri riešení úlohy môžete predpokladať, že existuje funkcia  $\text{autobus}(x,y,s)$ , ktorá vykrelí autobus na súradniciach  $[x, y]$  (počítané od  $[1, 1]$ ) a  $s$  je jeden z dvoch možných stavov: Funguje alebo Pokazený. Ak by sa vám takéto zmýšľanie zdalo priveľmi abstraktné, na stránke [http://www.ksp.sk/ksp/zadania/prilohy/ps241/znieny\\_autobus/](http://www.ksp.sk/ksp/zadania/prilohy/ps241/znieny_autobus/) sú implementované funkcie pre rôzne prostredia. Ak tam funkcia, ktorú potrebujete, nie je, predpokladajte, že existuje presne takáto pre váš programovací jazyk (v najhoršom si ju napíšete :-)). Za to, čo robí kód na tejto stránke, sa body neprideľujú (ani za vylepšenie tohto kódu). Sústreďte sa na zadanie úlohy, nie na pekný autobus.

## 3. Zúfalí Zimbabwejskí Zvedavci

kat. Z; 15 bodov

V Zimbabwe sa konali voľby. Každý normálny Zimbabwejčan odvolil, ako sa patrí, a teraz si už spokojne sedí doma a popíja Káčko<sup>4</sup>. Iba v klube Zimbabwejských zvedavcov je stále rušno. Zvedavci chcú čo najskôr vedieť všetky rôzne koalície, ktoré si môžu ich politici dohodnúť. Každú chvíľu vykrikuje nejaký zvedavec: „A ZDKU, HZDZ, ZNS a ZRZ? Takú ešte nemáme.“ a druhý mu odpovie: „Ale veď ZRZ je tam úplne zbytočne, lebo ZDKU, HZDZ a ZNS majú väčšinu aj bez nich. A okrem toho ZDKU nechce byť vo vláde spolu s HZDZ.“ No a takto sa dohadujú už druhý deň.

Ich úloha možno vyzerá ľahko a banálne. Ale to asi neviete, že v Zimbabwe má okrem pár väčších strán svoju stranu aj každý druhý kmeň. Pre každú stranu je navyše známy zoznam strán, s ktorými odmieta byť vo vláde. Niet divu, že Zimbabwejskí zvedavci potrebujú vašu pomoc.

**Úloha:** Na vstupe dostanete číslo  $N$  – počet strán, ktoré sa dostali do parlamentu. Potom bude nasledovať  $N$  riadkov. Každý riadok reprezentuje jednu stranu. V riadku  $i$  je počet kresiel v parlamente, ktoré získala  $i$ -ta strana (pre jednoduchosť predpokladajte, že strany sú očíslované od 1 po  $N$ , nemusíte uvažovať zložité názvy strán) a potom nasleduje zoznam čísel strán, s ktorými  $i$ -ta strana odmieta byť v koalícii. Tento zoznam je vždy zakončený číslom 0.

Vašou úlohou je vypísať všetky možné koalície. Koalícia musí mať nadpolovičnú väčšinu kresiel, žiadna jej strana nesmie mať výhrady proti žiadnej inej a žiadna strana v nej nesmie byť zbytočná. (Zbytočná je vtedy, ak by aj bez nej mali zvyšné strany nadpolovičnú väčšinu.)

---

<sup>4</sup>Zimbabwe je druhý najväčší odberateľ Kofoly na svete

**Príklad:****Vstup:**

8  
47 3 0  
42 0  
31 0  
8 2 3 0  
17 2 0  
3 3 8 7 0 12 2 0  
18 2 0

**Výstup:**

1 2 6  
5 1 8 7  
8 5 1 4  
(Všimnite si, že 1 a 2 má spolu 89 hlasov, čo je v tomto prípade málo.  
Takisto by vládu mohla tvoriť koalícia 5 1 8 7 4, ale strana č. 4 je tam zbytočná.)

..... Tu začínajú zadania KSP-O .....

**4. Zlostný generál**

kat. Z a O; 15 bodov

„Vojaci pó-zor! Vy polointeligentní tubíci, do dvojradu nastúpiť!“, zreval major Major. „Po-malé to bolo, všetci 20 klikov! Teraz!“. Tak chudákom vojakom neostalo nič iné, ako spraviť 20 klikov. Keď skončili, tak sa všetci rýchlo postavili do dvojstupu, nech im ten magor (major (Major)) nedá klikovať ešte raz.

„Takže ja som magor? Ďalších 50!“. Chudáci vojaci. Keď skončili, tak sa ešte rýchlejšie postavili do dvojradu. „Ták, vy predpotopné krysy, myslíte si, že som s vami skončil? Teraz sa v každom rade utriedte od najvyššieho po najnižšieho!“ Tak sa vojaci začali triediť. Bolo v tom dosť zmätkov, no nakoniec si vymysleli systém, pri ktorom sa menili vždy dvaja susední, a tak sa im to napokon podarilo. Trvalo to však tak dlho, že na nich potom major Major zreval: „Vy hyperbolické lamy! Tomu hovoríte rýchle triedenie? 400 klikov každý!“ Tak to už bolo veľa. Keď skončili, boli všetci takí unavení, že ďalšie kliky by nezvládli. „A teraz, vy rýchloplaché veveričky, nech vystúpi ten, čo má z vás strednú výšku...“

**Úloha:** Predpokladajte, že v pamäti máte uložené dve rovnako veľké polia  $A[1..N]$  a  $B[1..N]$ . Každé z týchto polí je utriedené od najmenšieho prvku po najväčší. Vašou úlohou je napísať čo najefektívnejší algoritmus, ktorý nájde medián spomedzi všetkých  $2N$  prvkov. (Teda taký prvok, ktorý by bol na  $N$ -tom mieste, keby sme všetkých  $2N$  prvkov dali do jedného poľa a toto pole utriedili.)

Snažte sa, aby Váš program bol čo najefektívnejší. Načítanie vstupu a polia  $A$  a  $B$  neuvažujte v odhade časovej a pamäťovej zložitosti.

**Príklad:****Vstup:**

$N = 5$   
 $A = \{1, 3, 6, 8, 12\}$   
 $B = \{3, 5, 10, 24, 30\}$

**Výstup:**

6

**5. Zase raz Kapingamarangi**

kat. Z a O; 15 bodov

Počuli ste už o Kapingamarangi? Že nie? Môžete ľutovať. Taký krásny atol ste určite ešte nevideli. A viete, že majú aj vlastný jazyk? Napríklad také slovo „mahina“ znamená mesiac. Plánujem tam ísť na najbližšiu dovolenku.

Odporúčaný spôsob, ako okolité atoly obdivovať, je ísť so sprievodcom na okružnú plavbu. Vlastne ani nie tak okružnú ako obdĺžnikovú. (To viete, v okolí Kapingamarangi je totiž  $\pi$  rovné štyrom.)

Mapu súostrovia si môžeme predstaviť ako štvorcovú sieť. Niektoré políčka obsahujú miesta vskutku nádherné, na ostatných je zase neuveriteľná nuda a otrava. (Ani len plávať tam nemôžem ísť, mám totiž panickú hrôzu zo zlatých rybičiek.) Každé nádherné miesto

zvýši môj zážitok o 1 a každé nudné miesto zníži môj zážitok o 1. A ja by som, pochopiteľne, chcel mať čo najväčší zážitok.

**Úloha:** Mapu súostrovia máme zadanú ako maticu  $A$  s  $R$  riadkami a  $S$  stĺpcami, ktorej prvky nadobúdajú hodnotu buď 0 (nudné miesto) alebo 1 (nádherné miesto).

Nech  $O$  je nejaký obdĺžnik v matici  $A$ . Nech  $P_1(O)$  je počet jednotiek a  $P_0(O)$  počet núl na jeho stranách. Potom rozdiel  $P_1(O) - P_0(O)$  hovorí, ako veľký je zážitok z plavby po obvodě obdĺžnika  $O$ . Nech  $Z$  je najväčší zážitok, ktorý vieme vhodnou voľbou obdĺžnika dosiahnuť.

Napište program, ktorý pre zadanú mapu vypočíta hodnotu  $Z$ , ako aj počet  $P$  rôznych obdĺžnikov, pre ktoré takýto zážitok dosiahneme.

**Príklad:**

**Vstup:**

$R = 7, S = 6$

1 1 1 1 1 0

1 0 0 0 1 0

1 0 0 0 1 0

1 1 1 1 1 0

1 0 0 0 0 0

1 0 0 0 1 0

1 1 1 1 1 0

**Výstup:**

18 1

..... Tu končia zadania KSP-Z .....

## 6. O Novom Jorku a teroristoch

kat. O; 15 bodov

Počuli ste? Že zas mali byť nejaké teroristické útoky, že zas mali padnúť nejaké tie mrakodrapy v nejakom tom Novom Jorku? S tým je odteraz koniec, povedala si miestna vláda. Rozhodla sa, že všetky mrakodrapy dá obaliť hrubou vrstvou pružnej gumy, a navyše bude evidovať jednotlivé zásahy. (To aby vedeli, kedy kam treba dodať novú vrstvu gumy.) Evidovať zásahy však nie je také ľahké, lebo jediné meracie zariadenie, ktoré majú, vie určiť iba výšku, v akej zamerané lietadlo letí, nič viac.

**Úloha:** Napište program, ktorý bude zaznamenávať lietanie lietadiel nad Novým Jorkom. Nový Jork si zjednodušíme na priamku, na ktorej je niekoľko budov. Z ľavej strany na pravú budú v rôznych výškach prelietavať lietadla, rovnobežne so zemou. Ako ste iste intuitívne pochopili, lietadlo narazí do prvej budovy, ktorá mu stojí v ceste (čiže má aspoň takú výšku, ako je výška letu lietadla). Po náraze lietadla budova ostáva na mieste bez akejkoľvek zmeny. Váš program by mal vedieť spracúvať nasledujúce požiadavky:

- Pre zadané lietadlo vypočítať, do ktorej budovy narazí (prípadne vypísať, že lietadlo preletí)
- Zaznamenať si, že sa zmenila výška niektorej budovy. (Napríklad postavili na vrchu televíznu vežu, prípadne sa nejaký milionár odsťahoval aj so svojimi tromi poschodiami budovy.)

Na vstupe je číslo  $N$  – počet budov v Novom Jorku. Na začiatku sú všetky domy vysoké 0 (naozaj, aj Nový Jork voľakedy vyzeral ako Bratislava). Nasleduje nešpecifikovaný počet riadkov troch typov:

- $1X$  – Letí lietadlo vo výške  $X$ , kde  $X$  je kladné celé číslo.
- $2XY$  –  $X$ -tá budova zľava mení svoju výšku na hodnotu  $Y$ . Budovy číslujeme od jedna.
- $0$  – Minuli sa lietadlá, na dnes končíme.

Pre každý riadok vstupu začínajúci jednotkou vypíšte jedno číslo – číslo budovy, do ktorej dané lietadlo narazí. Nič iné na výstup nepíšte.

**Príklad:****Vstup:**

$N = 7$   
 1 1  
 2 3 5  
 1 3  
 2 4 7  
 1 6  
 2 4 1  
 1 6  
 0

**Výstup:**

Preleti  
 3  
 4  
 Preleti

## 7. O VMVČ

kat. O; 15 bodov

Neviem, či poznáte skratku VMČ – znamená to „veľmi márnotravný človek“. Takíto ľudia majú dostatok peňazí a dávajú to patrične najavo – v obliekaní, vo vystupovaní, v stravovaní a dokonca aj v doprave! Takí VMČáci majú nielenže dostatok peňazí, ale aj času (pretože čas sú peniaze), a tak si môžu dovoliť nechodiť nikam priamo. Vždy, keď idú (vlastne letia) z jedného mesta do druhého, spravia si nejakú zachádzku a obzrú si po ceste aj iné mestá.

Medzi VMČákmi v tomto dokonca panuje istá rivalita. Existuje hierarchia VMČákov podľa toho, akú veľkú zachádzku sú schopní VMČáci pri ceste urobiť. VMČák je  $k$ -MČák, ak pri každej ceste letí aspoň  $k$  letmi (pochopte, čím väčšie  $k$ , tým uznávanejší). Tak napr. 3-MČák pri každej ceste nasadne aspoň trikrát do lietadla.

Samozrejme, ani medzi VMČákmi nie sú všetci „dokonalí“ – niektorí na to proste nemajú a stávajú sa z nich VMVČáci (veľmi márnotravné *vyzerajúci* ľudia).  $k$ -MVČák síce pri každej ceste spraví aspoň  $k$  letov, avšak lety dni predtým dôkladne vyberá tak, aby minul čo najmenej (dokonca aj za cenu toho, že po ceste prejde viackrát tým istým mestom).

**Úloha:** Dané je číslo  $n$  – počet miest,  $k$  – rád VMVČáka a tabuľka  $n \times n$  cien, pričom číslo v  $i$ -tom riadku a  $j$ -tom stĺpci znamená, koľko stojí let z mesta  $i$  do mesta  $j$  (okrem letov z  $i$ -teho späť do  $i$ -teho mesta – tie seriózní VMVČáci nepočítajú). Zostrojte obdobnú tabuľku cien, ktorá bude hovoriť, koľko stojí  $k$ -MVČáka cesta z každého mesta do každého iného. To znamená, pre každú dvojicu miest spočítajte cenu najlacnejšej cesty pozostávajúcej z aspoň  $k$  letov.

**Príklad:****Vstup:**

$n = 4, k = 2$   
 - 1 3 4  
 1 - 2 7  
 2 5 - 3  
 4 1 1 -

**Výstup:**

- 3 3 6  
 3 - 4 5  
 4 3 - 6  
 2 3 3 -

Všimnite si, že niekedy sa oplatí spraviť viac ako 2 lety.

..... Tu začínajú zadania KSP-T .....

## 8. Otravná aritmetická postupnosť

kat. O a T; 15 bodov

Aritmetická postupnosť<sup>5</sup> je skoro všade. Stretnete ju v obchode, na maturite, v zadaniach KSP, proste na každom kroku. Nech si zoberiete ľubovoľnú postupnosť celých čísel, určite v nej nejakú aritmetickú postupnosť nájdete. Nevyhnete sa jej. Alebo žeby predsa?

<sup>5</sup>  $n$ -členná aritmetická postupnosť s diferenciou  $d$  je postupnosť  $a_0, a_0 + d, a_0 + 2d, \dots, a_0 + (n-1) \cdot d$

**Úloha:** Postupnosť  $N$  celých čísel je zlá, ak vieme vyškrtnúť najviac  $N - 3$  z nich tak, aby tie, čo zostanú, tvorili aritmetickú postupnosť. Inými slovami, zlé postupnosti sú tie, ktoré obsahujú aritmetickú podpostupnosť dĺžky aspoň tri.

Na vstupe je číslo  $N$ . Nasleduje postupnosť  $N$  rôznych celých čísel. Napíšte program, ktorý tieto čísla preusporiada tak, aby výsledná postupnosť nebola zlá. Ak sa to nedá, podajte o tom vhodnú správu.

**Príklad:**

**Vstup:**

$N=6$

14 6 4 5 12 10

**Výstup:**

4 6 5 14 10 12

..... Tu končia zadania KSP-O .....

## 9. Trpasličie hry

kat. T; 15 bodov

V jednom trpasličom kráľovstve práve teraz oslavujú 47. narodeniny kráľa. Pri tejto príležitosti sa organizujú všelijaké súťaže. Súťaží sa napríklad v hode banánom, v kopaní hrobov, atď. Najťažšou súťažou je však „Hádanie slov“.

Táto súťaž prebieha vo veľkej štvorcovej miestnosti veľkosti  $N \times N$  metrov. Miestnosť je rozdelená na  $N \times N$  štvorcov so stranou dĺžky 1. V každom štvorci je napísané práve jedno veľké písmeno. Štvorec v severozápadnom rohu miestnosti má súradnice  $(1, 1)$ , protiľahlý štvorec je  $(N, N)$ .

Na začiatku súťaže sa postupne niekoľko trpaslíkov po miestnosti prejde. Každý z nich musí začať v štvorci so súradnicami  $(1, 1)$  a skončiť v štvorci so súradnicami  $(N, N)$ . V každom kroku sa trpaslík môže pohnúť buď o štvorec na východ, alebo o štvorec na juh. Navyše trpaslík nesmie prekročiť uhlopriečku (ale môže na ňu stúpiť). Ak sa teda v prvom kroku rozhodne ísť na juh, musí zostať v časti miestnosti, ktorá je južne od uhlopriečky.

Každý trpaslík si zapamätá postupnosť písmen, ktoré počas svojej cesty videl na zemi (písmen bude  $2N - 1$ ). Trpaslíkov pôjde po miestnosti presne toľko, koľko je všetkých možných ciest, a navyše žiadni dvaja trpaslíci nepôjdu po rovnakej ceste.

Na konci sa trpaslíci zoradia abecedne podľa toho, aké slovo počas cesty uvideli. Úlohou súťažiacich je povedať, ktoré bude  $K$ -te slovo v zoradenej postupnosti. Keďže kráľ hrá túto hru rád, ale už nezvláda rýchlo rozmýšľať, poprosil vás, aby ste mu pomohli.

**Úloha:** Na vstupe sú čísla  $N$  a  $K$ . Nasleduje matica písmen s rozmermi  $N \times N$ , udávajúca písmená napísané na podlahe miestnosti. Vašou úlohou je nájsť  $K$ -te slovo v zoradenej postupnosti trpaslíkov.

**Príklad:**

**Vstup:**

$N = 3, K = 3$

DAA

BDE

AAD

**Výstup:**

DBAAD

(Cesty sú DAAED, DADED, DBAAD, DBDAD, tretia je DBAAD.)

## 10. TurboTrúba

kat. T; 15 bodov

Tomáš má trúbu. A nie hocijakú, ale novú TurboTrúbu s USB 2.0. Keď ju má zapojenú do počítača, stačí kliknúť a trúba trúbi.

Jedného večera osvietilo Tomáša, že pomocou trúby by si mohol spraviť lokálnu sieť s kamarátom Tónom, ktorý býva o tri poschodia nad ním. Tóno si jednoducho kúpi mikrofón a Tomášov počítač bude Tónovmu posilať údaje trúbením.<sup>6</sup>

Trúba vie vydať dva rôzne zvuky: „tú“ a „húúú“. (Zvuk „húúú“ je dvakrát taký dlhý ako zvuk „tú“.)

Keďže Tomáš už čo-to pochytil z informatiky, vie, že „tú“ a „húúú“ môžu predstavovať bity, ktoré počítač na opačnom konci ľahko pochopí. Avšak takýto prenos je dosť pomalý (nehovoriac o ušiach susedov bývajúcich medzi ním a Tónom), preto bude treba prenášané správy efektívne kódovať.

Tomáša najviac zaujíma prenos slovenského textu. Rozhodol sa, že potrebuje kódovanie, ktoré bude mať nasledujúce vlastnosti:

- Každé písmeno abecedy zakódujeme ako nejakú postupnosť „tú“ a „húúú“.
- Žiadna z týchto postupností nie je prefixom (začiatkom) žiadnej inej.

Rozmyslite si, že druhá podmienka zabezpečuje, že zakódovanú vetu budeme môcť jednoznačne dekodovať, a to dokonca v lineárnom čase od jej dĺžky.

Samozrejme, kód, ktorý by písmeno „a“ kodoval ako postupnosť 47 „tú“, by bol dosť nanič. A tu už sa dostávate k slovu vy.

**Úloha:** Napíšte program, ktorý dostane na vstupe pre každé písmeno abecedy pravdepodobnosť, s akou sa vyskytuje v texte, a vyrobí optimálnu kódovaciu tabuľku.

Ako povedať, nakoľko dobrá je kódovacia tabuľka? Predstavte si, že vygenerujeme dlhý text, ktorý zodpovedá zadaným pravdepodobnostiam výskytu znakov, a tento text zakódujeme. Tabuľka je tým lepšia, čím kratší bude výsledný kód.

Pomôcka: Ak neviete s úlohou pohnúť, zamyslite sa najskôr nad jednoduchšou verziou, v ktorej trúba vydáva dva rovnako dlhé zvuky „tút“ a „píp“.

---

<sup>6</sup>Komunikáciu opačným smerom (ktorá s našou úlohou nesúvisí) už mali dávno vyriešenú. Uvedme len tolko, že prenosovým médium boli krúžky na záclonu, spúšťané po špagáte.