Sbírka úloh z programování

Dana Töpflerová

Pavel Töpfler

EDUCA '99

Grada

Dana Töpferová, Pavel Töpfer

Sbírka úloh z programování

EDUCA '99

Dana Töpferová, Pavel Töpfer

Sbírka úloh z programování

ISBN 80-85424-99-1

Obsah

Úvod

Tato kniha je sbírkou 400 úloh z programování různého charakteru a různé obtížnosti. Svou strukturou a zvolenou náročností odpovídá základnímu kursu programování v jazyce Pascal. Zadání většiny úloh však nijak nepodmiňuje volbu programovacího jazyka, takže sbírku je možné využít i při výuce programování založené na jiném programovacím jazyce.

Všechny úlohy jsou rozděleny do dvou oddílů, které v knize tvoří hlavní kapitoly II. a III. V kapitole II. jsou soustředěny jednodušší úlohy, u nichž není příliš obtížné nalézt správný algoritmus řešení a vhodně zvolit datové struktury. Tyto úlohy jsou orientovány na výuku jednotlivých rysů programovacího jazyka a podle toho jsou také rozčleněny do několika podkapitol. V rámci každé podkapitoly jsou uspořádány tak, aby obsahově podobné úlohy byly vždy pohromadě. Tam, kde je to možné, jsou skupiny podobných úloh navíc ještě uspořádány podle obtížnosti.

Úvodní podkapitola nazvaná Nejjednodušší úlohy obsahuje takové úlohy, při jejichž řešení nejsou zapotřebí žádné řídicí struktury programovacího jazyka. Jde jen o vyhodnocení jednoduchého aritmetického výrazu, a uvedené úlohy pouze ukazují možnosti volit různá slovní zadání "ze života". Podobná je i další podkapitola. Je věnována úlohám, při jejichž řešení potřebujeme podmíněné příkazy, ale ještě ne cykly. Zbývající části II. kapitoly obsahují úlohy na procvičení dalších rysů programovacího jazyka: cykly a pole, podprogramy, soubory a dynamické datové struktury. Tyto úlohy jsou již algoritmicky zajímavější. Zkušenosti z výuky ukazují, že při jejich řešení často dochází k různým opomenutím a chybám. Řada úloh je zde formulována poněkud "uměle", jsou to vlastně jen matematická zadání různých technických manipulací s čísly nebo znaky. To umožňuje soustředit se při výuce na důkladné procvičení a úspěšné zvládnutí jednotlivých prostředků programovacího jazyka. Souběžně s tím je samozřejmě možné řešit obsahově "hezčí" úlohy přiměřené obtížnosti vybrané ze III. kapitoly této sbírky.

Kapitola III. obsahuje většinou o něco náročnější příklady, které jsou zajímavé také z věcné stránky a mnohdy i z hlediska praktické použitelnosti. Je rovněž rozdělena do několika podkapitol, tentokrát podle tématického zaměření úloh. Uvnitř jednotlivých podkapitol jsou pak úlohy dále uspořádány podle věcné příbuznosti a podle obtížnosti. Náročnost úloh zahrnutých do této kapitoly je velmi rozmanitá, od poměrně snadných až po dosti obtížné. Při řešení úloh z kapitoly III. se předpokládá předchozí zvládnutí základních prostředků programovacího jazyka. U každé úlohy je nutné navrhnout algoritmus řešení a zvolit vhodné datové struktury a prostředky jazyka, pomocí nichž lze algoritmus naprogramovat.

V samostatné závěrečné části knihy (kapitola IV.) jsou uvedeny stručné návody k řešení vybraných úloh. Všechny úlohy ve sbírce jsou průběžně číslovány, takže vyhledání návodu k řešení zvolené úlohy je velmi snadné. Některé návody představují stručný nástin algoritmu řešení, jiné třeba jen doporučují výhodnou volbu datové struktury nebo upozorňují na nebezpečí častých chyb.

Kniha je určena každému, kdo chce zvládnout základy programování a potřebuje zásobu různých příkladů na procvičování. Najde uplatnění při výuce programování jak na středních školách, tak i v základním kursu programování na vysokých školách a v kursech jiných. K řešení úloh nejsou zapotřebí žádné předběžné znalosti z jiných oblastí, pouze v některých úlohách se využívají základy středoškolské matematiky.

Příklady zařazené do sbírky jsou částečně původní, částečně pocházejí z mnoha různých zdrojů. Některé vznikly úpravou úloh z jiných sbírek a učebnic, jiné vycházejí z osobních "zásob příkladů" těch pracovníků katedry kybernetiky a informatiky MFF UK v Praze, kteří se podílejí nebo v minulosti podíleli na výuce základního kursu programování na matematicko-fyzikální fakultě Univerzity Karlovy.

Základy programování

Nejjednodušší úlohy

Tento úvodní oddíl obsahuje nejjednodušší úlohy, jaké má vůbec smysl programovat. K jejich řešení nepotřebujeme znát, jak se v programovacím jazyce zapisují a vyhodnocují podmínky a jak vypadá podmíněný příkaz. Vystačíme se znalostmi celočíselných a reálných proměnných, zápisem aritmetického výrazu a příkazy vstupu a výstupu v základní podobě.

1. Průměrná denní teplota se počítá jako aritmetický průměr teplot naměřených v 6 hodin, ve 12 hodin a v 18 hodin, přičemž údaj naměřený v 18 hodin se do průměru započítává dvakrát. Spočítejte průměrnou denní teplotu z údajů o naměřených teplotách v 6 hodin, ve 12 hodin a v 18 hodin.

2. Zadanou rychlost v m/s převeďte na km/h.

3. Časový údaj zadaný ve dnech a hodinách převeďte na hodiny.

4. Časový údaj zadaný v sekundách převeďte na hodiny, minuty a sekundy.

5. Vytiskněte zadaných pět čísel v opačném pořadí, než v jakém jsou zadána na vstupu.

6. Je dán plánovaný počet výrobků a počet výrobků, které podnik opravdu vyrobil. Na základě těchto dvou hodnot určete, na kolik procent podnik splnil svůj plán výroby.

7. Na základě zadané výše vkladu (v Kčs) a roční úrokové míry (v procentech) spočítejte, jaký bude zůstatek na účtu na konci roku.

8. Spočítejte délku přepony pravoúhlého trojúhelníka, jsou-li dány délky jeho odvěsen.

9. Spočítejte vzdálenost dvou bodů v rovině, jsou-li dány souřadnice každého z nich.

10. Je dán poloměr koule. Spočítejte její objem a povrch podle známých matematických vzorců.

11. Je dán poloměr podstavy a výška nádoby tvaru válce (v centimetrech). Vypočítejte objem vody (v litrech), která se vejde do nádoby.

12. Spočítejte výšku rotačního kužele, je-li zadán jeho objem a poloměr podstavy.

13. Jsou zadány hodnoty dvou odporů v ohmech. Určete hodnotu celkového odporu při jejich sériovém a paralelním zapojení.

14. Jsou dána množství dvou roztoků téže látky (v litrech) a jejich koncentrace (v procentech). Spočítejte koncentraci roztoku, který vznikne jejich smícháním.

15. Pro zadaný čas (v sekundách) určete, jakou dráhu urazí za tento čas těleso padající volným pádem.

16. Nádrž má čtvercové dno o straně velikosti A. Přitéká do ní voda rychlostí N litrů za sekundu. Na základě vstupních údajů A (v metrech) a N (v litrech za sekundu) spočítejte, jak dlouho musí voda přitékat, aby hladina stoupla o dva metry.

17. Do soutěže se přihlásilo N hráčů. Určete, kolik zápasů je třeba sehrát, aby hrál každý s každým jednou. Počet hráčů N je zadán na vstupu.

Podmíněné příkazy

Úlohy obsažené v tomto oddílu knihy vyžadují znalost zápisu a vyhodnocování podmínek a použití podmíněného příkazu v programu. Nejsou zapotřebí žádné datové struktury ani cykly.

18. Určete, zda dvě čísla x, y zadaná na vstupu splňují nerovnost $4x + 3 < 5y - 1$.

19. Jsou dány souřadnice bodu v rovině. Zjistěte, zda tento bod leží na přímce $3x + 2y - 3 = 0$.

20. Jsou dána dvě kladná čísla, která představují délky stran obdélníka v centimetrech. Zjistěte, zda se jedná o čtverec, a vytiskněte zprávu ve tvaru
"Čtverec má délku strany ... cm"
nebo "Obdélník má rozměry ... cm x ... cm".
V případě obdélníka uveďte nejprve délku jeho delší strany.

21. Jsou známy naměřené údaje o dvou autech. První z nich ujelo trasu s_1 za čas t_1 a druhé trasu s_2 za čas t_2. Délky ujetých tras jsou dány v kilometrech a časy v hodinách. Na základě těchto čtyř vstupních údajů spočítejte průměrné rychlosti obou aut, porovnejte je a vypište zprávu, které auto je rychlejší a o kolik (např. ve tvaru "Průměrná rychlost prvního auta je větší o 10 km/h.").

22. Do nádoby ve tvaru válce o poloměru R a výšce H chceme nalít X litrů vody. Zjistěte, zda se všechna voda do nádoby vejde. Jestliže ano, spočítejte, jak vysoko ode dna nádoby bude hladina vody. Na vstupu jsou rozměry válce R, H zadány v centimetrech a množství vody X v litrech.

23. Střecha budovy má plochu 200 m². K nátěru je možné použít jeden ze dvou druhů barev. U každého druhu známe spotřebu barvy v kilogramech na m² plochy a cenu barvy v Kčs za 1 kilogram barvy. Určete, kterou barvu je výhodnější použít, aby nátěr celé střechy vyšel levněji. Dále určete, o kolik Kčs bude zvolený nátěr levnější než ve druhém případě. Je-li nátěr oběma druhy barvy stejně drahý, doporučte použít první z nich.

24. Jsou dány výsledky třikrát opakovaného fyzikálního měření jisté veličiny a dále předpokládaná výsledná hodnota, která měla být naměřena. Určete, kolikrát se naměřená hodnota odchýlila od očekávané hodnoty více než o 10 procent.

25. Pro danou dvojici čísel x, y spočítejte hodnotu výrazu 1/(x.y).

26. Zadaná tři čísla uspořádejte vzestupně podle jejich absolutních hodnot (tzn. jako první vytiskněte to z čísel na vstupu, které má nejmenší absolutní hodnotu, atd.).

27. V průběhu jednoho roku jste si zaznamenali náklady na provoz domácnosti v každém čtvrtletí. Na základě zaznamenaných údajů (tj. čtyř čísel) vypište zprávu, ve kterém čtvrtletí byly náklady největší a kolik činily.

28. Jsou dána čtyři celá čísla. Určete největší a nejmenší z nich.

29. Zjistěte, zda daná tři čísla mohou být stranami trojúhelníka.

30. Ověřte, zda daná čtveřice čísel má následující vlastnost: součet libovolných tří z nich je větší než zbývající čtvrté číslo.

31. Jsou dány hodnoty koeficientů A, B lineární rovnice Ax+B=0. Vyřešte tuto rovnici.

32. Jsou dány hodnoty koeficientů A, B, C kvadratické rovnice $Ax^2+Bx+C=0$. Vypočítejte reálné kořeny této rovnice.

33. Je dán počet dní v měsíci (tj. číslo z rozmezí 28 až 31) a informace, na který den v týdnu připadá první den v měsíci (tato informace je ve tvaru pořadového čísla: pondělí=1, úterý=2, ..., neděle=7). Zjistěte, kolik je v daném měsíci pátků.

34. Je dán počet dní v měsíci a informace, na který den v týdnu připadá první den v měsíci (tato informace je ve tvaru pořadového čísla: pondělí=1, úterý=2, ..., neděle=7). Zjistěte, kolik je v daném měsíci pracovních dní (pracovní dny jsou pondělí až pátek).

Jednoduché cykly

Tento oddíl obsahuje úlohy na první procvičení práce s cykly. Při řešení každé ze zde uvedených úloh stačí použít vedle podmíněných příkazů jen jediný cyklus. Nepotřebujeme používat ani žádné datové struktury (pole, záznamy, soubory apod.). Obtížnějším úlohám na procvičení cyklů bude věnován celý následující oddíl knihy.

V některých úlohách je úkolem přečíst ze vstupu a nějak zpracovat posloupnost čísel. U těchto úloh je možné vytvářet různé modifikace zadání podle toho, jak má být čtení čísel ukončeno. Nejjednodušším případem je situace, kdy je v programu pevně dáno konstantou, kolik čísel se má zpracovat. Obecnější řešení předpokládá, že na vstupu je nejprve zadáno kladné celé číslo N udávající, kolik čísel bude za ním na vstupu následovat a má být zpracováno. Jinou možností je, víme-li například, že všechna zpracovávaná čísla jsou kladná. Jejich posloupnost na vstupu potom můžeme zakončit třeba nulou nebo nějakým záporným číslem, které se už nezpracovává. Jindy známe určitou vlastnost posledního čísla, kterou se liší od všech předchozích. Tuto znalost využijeme na ukončení cyklu, přičemž poslední číslo ještě zpracujeme stejně jako všechna ostatní čísla. Vrátíme-li se k některé z těchto úloh ve chvíli, kdy již známe způsob práce se soubory, můžeme jednoduše číst a zpracovávat čísla ze vstupu až do vyčerpání vstupního souboru.

35. Ve škole je 19 tříd. Známe počty žáků v jednotlivých třídách (čísla daná na vstupu). Spočítejte počet všech žáků školy.

36. Ve škole je 19 tříd. Známe počty žáků v jednotlivých třídách (čísla daná na vstupu). Spočítejte průměrný počet žáků ve třídě.

37. Máme schované účtenky z několika nákupů a chceme zjistit, kolik jsme celkem utratili. Je dána posloupnost kladných celých čísel (ceny jednotlivých nákupů) zakončená nulou. Spočítejte součet těchto čísel (celková útrata).

38. Máme schované účtenky z několika nákupů a chceme zjistit, kolik jsme utratili průměrně při jednom nákupu. Je dána posloupnost kladných celých čísel (ceny jednotlivých nákupů) zakončená

nulou. Spočítejte aritmetický průměr těchto čísel (průměrná cena nákupu).

39. Je dána posloupnost 30 celých čísel. Spočítejte, kolik z nich je kladných, kolik záporných a kolik nulových.

40. Je dána posloupnost nenulových čísel zakončená nulou. Spočítejte, zda je mezi nimi více čísel kladných nebo záporných a o kolik.

41. Je dán seznam příjmů (kladná celá čísla) a výdajů (záporná celá čísla) za zvolené období. Posloupnost těchto čísel je zakončena nulou. Spočítejte celkové příjmy, celkové výdaje a výslednou platební bilanci (úspory nebo dluh).

42. Je dána posloupnost kladných celých čísel zakončená nulou. Určete, kolik je v ní lichých čísel dělitelných třemi.

43. Jsou dány hodnoty A, B a posloupnost kladných celých čísel zakončená nulou. Zjistěte, kolik z těchto čísel je z intervalu <A,B> (tj. větších nebo rovných A a zároveň menších nebo rovných B).

44. Je dána posloupnost 50 celých čísel. Spočítejte hodnotu aritmetického průměru těch čísel z posloupnosti, která jsou větší než nula a zároveň menší než sto.

45. Je dána posloupnost 50 celých čísel. Spočítejte aritmetický průměr všech čísel umístěných v posloupnosti na sudých pozicích (tzn. druhého, čtvrtého, ...) a aritmetický průměr čísel stojících na lichých pozicích v posloupnosti (tzn. prvního, třetího, ...).

46. Vyhodnoťte klasifikaci z jedné písemné práce ve třídě. Je dán počet žáků ve třídě a dále výsledná známka každého z žáků (známky 1, 2, 3, 4 a 5). Určete počty žáků klasifikovaných jednotlivými známkami a průměr celé třídy.

47. Jsou dány trojice čísel udávající prospěch jednotlivých žáků z českého jazyka, matematiky a angličtiny. Vstup je ukončen trojicí 0, 0, 0. Určete:
   a) průměrný prospěch třídy z angličtiny

   b) kolik žáků má jedničku z českého jazyka, kolik z matematiky a kolik z angličtiny
   c) kolik žáků má jedničku ze všech tří předmětů
   d) kolik žáků nemá žádnou čtyřku ani pětku

   Úlohu můžete řešit i jako čtyři samostatné úlohy.

48. Vstupními údaji jsou výše platů všech zaměstnanců podniku (kladná celá čísla). Jako poslední je uveden plat ředitele. Ředitel jako jediný má plat vyšší než 10.000,- Kčs. Zjistěte:
   a) počet zaměstnanců podniku
   b) průměrný plat všech zaměstnanců podniku
   c) o kolik je ředitelův plat vyšší než průměrný plat ostatních zaměstnanců
   d) rozdíl mezi nejvyšším (tj. ředitelovým) a nejnižším platem v podniku
   e) počet zaměstnanců s platem vyšším než 5.000,- Kčs

   Úlohu můžete řešit i jako pět samostatných úloh.

49. Vstupními údaji jsou výše platů všech zaměstnanců podniku (kladná celá čísla). Jako poslední je uveden plat ředitele. Ředitel jako jediný má plat vyšší než 10.000,- Kčs. Proveďte platovou úpravu, při níž budou všechny mzdy menší než 4.000,- Kčs zvýšeny o 500,- Kčs a všechny mzdy z rozmezí od 4.000,- do 4.500,- Kčs upraveny na 4.500,- Kčs. Ostatní platy zůstanou beze změn. Novou posloupnost platů všech zaměstnanců podniku vypište v původním pořadí.

50. Zadaná posloupnost kladných čísel představuje údaje o výši vkladů v Kčs na jednotlivých účtech. Posloupnost je ukončena libovolným záporným číslem. Spočítejte výši vkladů na účtech po uplynutí jednoho roku. Každý vklad má být zvýšen o 7% a nové stavy účtů vypsány v původním pořadí.

51. Je dána posloupnost dvojic kladných čísel. První z dvojice představuje vždy údaj o výši vkladu v Kčs a druhé číslo je úroková míra v procentech, podle které jsou na tento účet připisovány úroky. Vstup je ukončen dvojicí nul. Připište na všechny účty úroky po uplynutí jednoho roku. Každý vklad bude zvýšen podle své příslušné úrokové míry. Nové hodnoty vkladů na účtech vytiskněte v původním pořadí.

52. Na základě hodnoty počátečního vkladu (v Kčs), výše roční úrokové míry (v procentech) a doby uložení úspor (v letech) spočítejte, o kolik Kčs vzrostou vaše úspory.

53. Na začátku každého roku uložíme částku C Kčs na úrokovou vkladní knížku s ročním úrokem P%. Údaje C a P jsou zadány na vstupu. Zjistěte, kolik Kčs budeme mít na vkladní knížce za 10 let.

54. Na začátku každého roku uložíme částku C Kčs na úrokovou vkladní knížku s ročním úrokem P%. Údaje C a P jsou zadány na vstupu. Zjistěte, za kolik let naše úspory překročí částku X Kčs. Údaj X je rovněž zadán na vstupu programu.

55. Spočítejte hodnotu obecné mocniny x^n pro dané reálné číslo x a přirozené číslo n.

56. Spočítejte hodnotu obecné mocniny x^n pro dané reálné číslo x a celé číslo n.

57. Je dána vstupní hodnota N (počet zpracovávaných čísel) a za ní N čísel. Pro každé z těchto N čísel spočítejte a vytiskněte hodnotu výrazu $x^2 + 6x + 1$.

58. Vypočítejte funkční hodnoty funkce $f(x) = x^3 + 4x - 1$ pro všechny celočíselné hodnoty argumentu x z intervalu od 1 do 20. Výsledek vytiskněte ve tvaru tabulky obsahující odpovídající dvojice hodnot x, f(x).

59. Sestavte tabulku funkčních hodnot funkce $g(x) = x^2/2 - 1$. Vstupním údajem je interval definičního oboru $<A,B>$, v němž nás funkční hodnoty zajímají, a dále krok H, s nímž máme funkci g(x) tabelovat. To znamená, že máme spočítat hodnoty funkce g(x) postupně pro hodnoty argumentu x rovné A, A+H, A+2H, A+3H atd., dokud nepřekročíme horní mez B.

60. Tabelujte funkci h(x), tzn. vytiskněte dvojice hodnot x, h(x) pro všechna x od -10 do 10 s krokem 0,1. Funkce h(x) je dána předpisem:
$h(x) = -2x-2$ pro $x < -2$
$h(x) = x+4$ pro $-2 \leq x \leq 8$
$h(x) = -2x+28$ pro $x > 8$

61. Pro dané kladné celé číslo n spočítejte hodnotu n! (faktoriál čísla n), tj. součin všech kladných celých čísel od 1 do n. Předpokládejte, že program bude zpracovávat pouze tak malé vstupní hodnoty n, že výsledný n! nepřekročí rozsah povolených hodnot typu integer.

62. Spočítejte součet převrácených hodnot všech kladných celých čísel od 1 do daného N.

63. Spočítejte hodnotu kombinačního čísla (n nad k) pro daná nezáporná celá čísla n, k, kde n>k. Tato hodnota je definována výrazem $n \cdot (n-1) \cdot \cdots \cdot (n-k+1) / k!$

64. Je zadána věta zakončená tečkou. Zjistěte, kolikrát se v této větě vyskytuje písmeno 'A'.

65. Je zadána věta. Jednotlivá slova jsou oddělena vždy jednou mezerou, za posledním slovem je tečka. Určete, kolik je v této větě slov.

66. Je zadána věta. Jednotlivá slova jsou oddělena libovolným počtem mezer, za posledním slovem je tečka. Určete, kolik je v této větě slov.

67. Je zadána je věta zakončená tečkou. Zjistěte, kolik je v ní velkých písmen.

68. Je dáno přirozené číslo N následované posloupností N celých čísel. Nalezněte maximální hodnotu v této posloupnosti.

69. Ze zadaných 50 čísel vyberte a vytiskněte číslo s největší absolutní hodnotou.

70. Je dána posloupnost kladných celých čísel ukončená nulou. Určete, kolikrát se mezi danými čísly nachází největší z nich.

71. Je dáno přirozené číslo N následované posloupností N celých čísel. Nalezněte v této posloupnosti co největší číslo menší než 1000.

72. Ze zadaných 30 různých čísel vyberte takové, jehož hodnota je nejbližší číslu 100. Existují-li dvě různá čísla s minimální vzdáleností od čísla 100 (jedno větší než 100 a druhé menší než 100), uveďte jako výsledek obě.

73. Je dáno 30 navzájem různých celých čísel. Nalezněte a vypište druhé největší z nich.

74. Je dána posloupnost celých kladných čísel zakončená nulou. Určete největší a nejmenší číslo v dané posloupnosti.

75. Je dáno kladné celé číslo N a posloupnost N celých čísel. Určete druhou nejmenší hodnotu nacházející se v zadané posloupnosti čísel a počet jejích výskytů.

76. Je dáno kladné celé číslo N (N<30) a dále N naměřených hodnot jisté fyzikální veličiny. Určete, při kolikátém měření došlo k největší odchylce od aritmetického průměru všech naměřených hodnot. Jakou měla tato největší odchylka hodnotu?

77. Je zadáno nejprve číslo X a pak posloupnost kladných celých čísel uspořádaná vzestupně podle velikosti. Posloupnost je zakončena nulou. Zařaďte číslo X do posloupnosti na správné místo tak, aby byla celá posloupnost opět uspořádaná. Výslednou posloupnost vytiskněte.

78. Je dáno kladné celé číslo N a posloupnost tvořená N celými čísly. Zjistěte, zda je tato posloupnost rostoucí (tj. uspořádaná vzestupně podle velikosti, čísla navzájem různá).

79. Je dána posloupnost 100 celých čísel. Zjistěte a vytiskněte informaci o monotonii této posloupnosti, tj. zda je rostoucí, nerostoucí, klesající, neklesající, konstantní, nebo není monotónní.

80. Zjistěte, zda je dané kladné celé číslo prvočíslo.

81. Určete počet cifer daného přirozeného čísla.

82. Spočítejte ciferný součet daného přirozeného čísla.

83. Spočítejte součet těch cifer daného přirozeného čísla, které leží na lichých pozicích počítáno odzadu (tj. poslední, třetí od konce, atd.).

84. Spočítejte součet těch cifer daného přirozeného čísla, které leží na lichých pozicích počítáno odpředu (tj. první, třetí, atd.).

85. Zjistěte, kolikrát se v daném přirozeném čísle vyskytuje cifra 7.

86. Jsou dána přirozená čísla N a K. Určete K-tou cifru zprava v čísle N. Má-li číslo N méně než K cifer, oznamte vypsáním chybové zprávy, že úloha nemá řešení.

87. Jsou dána přirozená čísla N a K. Určete K-tou cifru zleva v čísle N. Má-li číslo N méně než K cifer, oznamte vypsáním chybové zprávy, že úloha nemá řešení.

88. K danému kladnému celému číslu N spočítejte číslo, které se liší od čísla N tím, že má zaměněnu první a poslední cifru.

89. K danému kladnému celému číslu N spočítejte číslo, které má stejné cifry jako N, ale v opačném pořadí.

90. Nalezněte největší mocninu dvojky, která je menší než dané kladné celé číslo N. To znamená, že hledáme takové číslo X tvaru 2^k, aby $X<N$ a přitom $2{\cdot}X\geq N$.

91. Převeďte daný dvojkový zápis čísla na dekadický.

92. Převeďte dané přirozené číslo do dvojkové soustavy.

93. V zadané posloupnosti nenulových čísel zjistěte délku co nejdelšího souvislého úseku tvořeného kladnými čísly. Posloupnost je na vstupu ukončena nulou.

94. Je dána posloupnost 100 celých čísel. Určete délku co nejdelšího souvislého rostoucího úseku, který je v ní obsažen.

95. V zadané posloupnosti nenulových čísel (čísla se mohou v posloupnosti libovolně opakovat) zjistěte délku co nejdelšího souvis-

lého úseku tvořeného stejnými čísly. Posloupnost vstupních údajů je ukončena nulou.

96. V zadané posloupnosti nenulových čísel (čísla se mohou v posloupnosti libovolně opakovat) zjistěte délku co nejdelšího souvislého úseku tvořeného stejnými čísly. Dále zjistěte počet takových úseků stejných čísel, které mají maximální délku. Posloupnost vstupních údajů je ukončena nulou.

97. V dané posloupnosti celých čísel zjistěte délku co nejdelšího souvislého úseku, který je aritmetickou posloupností. V aritmetické posloupnosti je rozdíl dvou po sobě jdoucích prvků konstantní, tj.

rozdíl prvního a druhého čísla se rovná rozdílu druhého a třetího čísla, rozdílu třetího a čtvrtého čísla, atd.

98. Zadaná posloupnost čísel je rozdělena nulami na úseky tvořené nenulovými čísly. Ukončena je dvěma nulami následujícími bezprostředně po sobě. Určete hodnotu největšího čísla v nejdelším úseku. Předpokládejte, že úseky nenulových čísel mají různé délky.

99. Zadaná posloupnost čísel je rozdělena nulami na úseky tvořené nenulovými čísly. Ukončena je dvěma nulami následujícími bezprostředně po sobě. Určete hodnotu největšího čísla v nejdelším úseku. Má-li více úseků nenulových čísel stejnou délku, bude výsledkem největší číslo z prvního úseku této maximální délky.

100. Zadaná posloupnost čísel je rozdělena nulami na úseky tvořené nenulovými čísly. Ukončena je dvěma nulami následujícími bezprostředně po sobě. Určete hodnotu největšího čísla v nejdelším úseku. Má-li více nenulových úseků stejnou délku, bude výsledkem maximum z největších čísel v úsecích maximální délky.

101. Exponenciální funkci e^x můžeme vyjádřit jako součet nekonečné mocninné řady

$$e^x = 1 + x/1! + x^2/2! + \dots + x^i/i! + \dots,$$

kde výraz i! ve jmenovateli zlomku představuje faktoriál čísla i, tj. součin všech kladných celých čísel od 1 do i. Dosadíme-li do

tohoto rozvoje funkce e^x hodnotu $x=1$, dostaneme předpis pro aproximaci hodnoty čísla e (tzn. výpočet přibližné hodnoty čísla e s libovolnou zvolenou přesností). Spočítejte hodnotu čísla e s přesností 0,001. Přesností 0,001 rozumíme, že rozdíl dvou po sobě jdoucích aproximací je menší než 0,001.

102. Hodnotu Ludolfova čísla π lze aproximovat nekonečnou řadou

$$\pi = 4 \cdot (1 - 1/3 + 1/5 - 1/7 + 1/9 - \dots).$$

Vypočítejte pomocí této řady hodnotu π s přesností 0.001. Přesností 0,001 rozumíme, že rozdíl dvou po sobě jdoucích aproximací je menší než 0,001.

Cykly a pole

Tato část sbírky je tvořena dalšími úlohami na práci s cykly. Na rozdíl od předchozího oddílu se zde již v řešení úloh objevuje více cyklů, ať už prováděných po sobě nebo vnořených do sebe. Při řešení většiny úloh je zapotřebí použít pole.

103. Určete počet různých cifer daného přirozeného čísla.

104. Určete počet výskytů jednotlivých písmen 'A' až 'Z' v dané posloupnosti znaků. Posloupnost je zakončena hvězdičkou.

105. Jsou dány dva řetězce znaků. Určete, zda je jeden permutací druhého, tj. zda je možné získat jeden řetězec z druhého nějakou změnou pořadí jeho znaků. Například řetězec 'ABCAC' je permutací řetězce 'CCAAB', zatímco řetězec 'BBAAC' ne.

106. Je dána posloupnost 50 čísel. Zjistěte, kolik z nich se rovná poslednímu z čísel na vstupu.

107. Je dáno kladné celé číslo N (N<100) a za ním N zpracovávaných čísel. Zjistěte, kolik ze zpracovávaných čísel je dělitelných sedmi. Vytiskněte nejprve tento počet, a pak vypište všechna zpracovávaná čísla, která jsou dělitelná sedmi.

108. Je dána posloupnost 30 čísel. Vypište nejprve dvojnásobky všech kladných čísel, a pak absolutní hodnoty všech nekladných čísel.

109. Je dána posloupnost 50 čísel. Vypište tato čísla v obráceném pořadí (tj. nejprve poslední, pak předposlední, atd.).

110. Je dána posloupnost 50 čísel. Vypište tato čísla uspořádaná do následujícího pořadí: první, třetí, páté,..., čtyřicáté deváté, druhé, čtvrté, ..., padesáté.

111. Je dána posloupnost 50 celých čísel. Uveďte, na kolikátém místě v posloupnosti se nachází nejmenší z čísel. Vyskytuje-li se tam vícekrát, vypište pozice všech jeho výskytů.

112. Je dána posloupnost 100 celých čísel. Přerovnejte čísla tak, aby na začátku posloupnosti byla všechna nezáporná (v libovolném pořadí) a za nimi všechna záporná (opět v libovolném pořadí). Upravenou posloupnost vypište.

113. Je dána posloupnost 100 celých čísel. Přerovnejte čísla tak, aby na začátku posloupnosti byla všechna nezáporná čísla v původním pořadí a za nimi všechna záporná čísla opět se zachováním jejich původního pořadí. Upravenou posloupnost vypište.

114. Je dána posloupnost kladných celých čísel ukončená nulou. Víme, že čísel je méně než 100. Vypočtěte jejich aritmetický průměr a zjistěte pořadové číslo vstupního údaje, který se nejvíce blíží zjištěnému aritmetickému průměru.

115. V pravidelných časových intervalech byla sledována jistá fyzikální veličina (např. rychlost pohybujícího se tělesa). Zpracujte posloupnost 50 naměřených hodnot této veličiny tak, že k ní spočítáte posloupnost klouzavých průměrů délky 3. Označíme-li naměřené hodnoty postupně a_1, a_2, ..., a_{50}, pak klouzavými průměry délky 3 rozumíme čísla $(a_1+a_2+a_3)/3$, $(a_2+a_3+a_4)/3$, ..., $(a_{48}+a_{49}+a_{50})/3$. Tato čísla se někdy používají místo původně naměřených hodnot k sestrojení grafu závislosti sledované veličiny na čase. Jejich použití přináší jisté "vyhlazení" získané závislostní křivky a snížení vlivu náhodných chyb měření.

116. Jsou dána dvě kladná celá čísla K a N, kde K<N, a dále N hodnot fyzikální veličiny, naměřených v pravidelných časových intervalech. Vypočítejte posloupnost klouzavých průměrů délky K sledované fyzikální veličiny. Označíme-li naměřené hodnoty postupně a_1, a_2, ..., a_N, pak klouzavými průměry délky K rozumíme čísla $(a_1+a_2+...+a_K)/K$, $(a_2+a_3+...+a_{K+1})/K$, ..., $(a_{N-K+1}+a_{N-K+2}+ ...+a_N)/K$.

117. Je dána posloupnost kladných celých čísel ukončená nulou, čísel je méně než 100. Zjistěte, zda se v dané posloupnosti nějaké číslo opakuje.

118. Je dána posloupnost 50 čísel, z nichž některá mohou být stejná. Určete, kolik různých hodnot se mezi vstupními čísly nachází.

119. Je dána posloupnost 50 čísel, z nichž některá mohou být stejná. Určete, kolik hodnot se mezi vstupními čísly nachází pouze jednou.

120. Je dána posloupnost 50 čísel, z nichž některá mohou být stejná. Určete, kolik hodnot se mezi vstupními čísly nachází alespoň dvakrát.

121. Je dána posloupnost 50 čísel, z nichž některá mohou být stejná. Určete, kolik hodnot se mezi vstupními čísly nachází přesně dvakrát.

122. Je dána posloupnost 50 celých čísel. Nalezněte nejmenší číslo, které je v ní obsaženo alespoň dvakrát.

123. Je dána posloupnost 50 celých čísel. Nalezněte nejmenší číslo, které je v ní obsaženo přesně dvakrát.

124. Je dána posloupnost 50 celých čísel. Určete délku nejdelšího souvislého symetrického úseku, který je v ní obsažen. Např. nejdelším souvislým symetrickým úsekem obsaženým v posloupnosti čísel 2, 9, 7, 4, 7, 9, 7, 2, 9 je úsek 9, 7, 4, 7, 9 délky 5.

125. Je dána posloupnost 50 celých čísel. Vyhledejte v ní nejdelší souvislý symetrický úsek. Je-li takových úseků více, vypište všechny souvislé symetrické úseky maximální délky.

126. Je dána posloupnost 50 celých čísel. Vyhledejte v ní nejdelší souvislý rostoucí úsek. Je-li takových úseků více, vypište všechny souvislé rostoucí úseky maximální délky.

127. Vypište kalendář na měsíc prosinec například ve tvaru:

Po	Út	St	Čt	Pá	So	Ne
		1	2	3	4	5
6	7	8	9	10	11	12

......

Vstupem programu je pořadové číslo toho dne v týdnu, na který připadne 1.12. (V uvedeném příkladě je 1.12. ve středu, na vstupu bylo tedy zadáno číslo 3.)

128. Je dáno datum ve tvaru den, měsíc. Určete pořadové číslo tohoto dne v roce. Předpokládejte, že rok není přestupný.

129. Je dáno datum ve tvaru den, měsíc, rok. Určete pořadové číslo tohoto dne v daném roce. Předpokládejte, že rok je z našeho století.

130. Na základě pořadového čísla dne v roce a údaje, zda se jedná o přestupný rok, vypočítejte a vytiskněte datum tohoto dne (den a měsíc).

131. Jsou dána dvě data ve tvaru den, měsíc, rok. Určete počet dní, který uplyne mezi těmito dvěma daty. Můžete se omezit na data z rozmezí od roku 1800 do roku 2100.

132. Je dáno datum ve tvaru den, měsíc, rok. Určete, na který den v týdnu toto datum připadne. Můžete se omezit pouze na data z tohoto století.

133. Je dán počet dětí, délka rozpočítadla a pořadové číslo dítěte, od kterého se začíná rozpočítávat. Určete pořadí, v jakém budou děti vypadávat z kola.

134. Obdélník je zadán souřadnicemi tří svých vrcholů, které tvoří pravoúhlý trojúhelník. Dále jsou dány souřadnice bodu X v rovině. Určete souřadnice takového bodu na obvodu obdélníka, který je nejbližší bodu X.

135. Je dáno kladné celé číslo N. Spočítejte prvních N prvočísel.

136. Je dáno kladné celé číslo N. Určete a vytiskněte všechna prvočísla menší než N.

137. Je dáno kladné celé číslo N. Nalezněte a vypište prvních N prvočíselných dvojic. Prvočíselnými dvojicí nazýváme dvojici prvočísel lišících se o 2 (např. prvočísla 11 a 13 tvoří jednu takovou dvojici prvočísel).

138. Je dáno kladné celé číslo N. Nalezněte a vypište všechny prvočíselné dvojice tvořené čísly menšími než N.

139. Rozložte dané kladné celé číslo na prvočinitele. Rozkladem čísla na prvočinitele rozumíme vyjádření čísla ve tvaru součinu prvočísel.

140. Je dána posloupnost celých kladných čísel ukončená nulou. Vyberte a vypište ta z nich, jejichž zápis v desítkové soustavě je tvořen ciframi ve vzestupném pořadí.

141. Je dána posloupnost celých kladných čísel ukončená nulou. Vyberte a vypište ta z nich, jejichž zápis ve dvojkové soustavě je symetrický.

142. Je dána posloupnost celých kladných čísel ukončená nulou. Vytvořte a vytiskněte stejně dlouhou posloupnost tvořenou přirozenými čísly, která mají oproti daným číslům převrácený zápis ve dvojkové soustavě (tj. cifry dvojkového zápisu nového čísla jsou v opačném pořadí než cifry dvojkového zápisu čísla původního).

143. Přirozené číslo se nazývá dokonalé, je-li rovno součtu všech svých (kladných) dělitelů s výjimkou sebe sama. Například 6 je dokonalé číslo, neboť $6=1+2+3$. Vytvořte program, který vytiskne všechna dokonalá čísla menší než dané N.

144. Přirozené číslo se nazývá dokonalé číslo 2. druhu, je-li rovno součinu všech svých (kladných) dělitelů s výjimkou sebe sama. Například číslo 6 je dokonalé číslo 2. druhu, neboť $6=1\cdot 2\cdot 3$. Vytvořte program, který vytiskne všechna dokonalá čísla 2. druhu menší než dané N.

145. Dvě kladná celá čísla A, B se nazývají spřátelená, jestliže číslo A je rovno součtu všech kladných dělitelů čísla B s výjimkou B samotného a číslo B je rovno součtu všech kladných dělitelů čísla A s výjimkou A samotného. Například 220 a 284 jsou spřátelená čísla, neboť

$$284=1+2+4+5+10+11+20+22+44+55+110 \text{ a}$$
$$220=1+2+4+71+142.$$

Vytvořte program, který vytiskne všechny dvojice spřátelených čísel menších než dané N, $N<1000$.

146. Jsou dána dvě kladná celá čísla A, B, $A<B$. Vytvořte program, který nalezne a vytiskne všechna prvočísla z intervalu <A,B> taková, že jejich zápis končí číslicí 1.

147. Vytiskněte všechna trojciferná přirozená čísla s daným ciferným součtem.

148. Jsou dána dvě kladná celá čísla N, K. Vytiskněte všechna kladná celá čísla menší než N, jejichž ciferný součet je roven K.

149. Jsou dána dvě kladná celá čísla N, K. Vytiskněte prvních N kladných celých čísel, jejichž ciferný součet je roven K.

150. Nalezněte všechna kladná celá čísla menší než dané N, která lze rozložit na součet dvou kvadrátů kladných celých čísel. Vytiskněte nalezená čísla i jejich rozklady.

151. Přirozené číslo se nazývá Armstrongovo, je-li rovno součtu třetích mocnin svých cifer (např. $153=1^3+5^3+3^3$). Nalezněte všechna Armstrongova čísla menší než dané celé číslo N.

152. Přirozené číslo se nazývá palindrom, je-li jeho dekadický zápis symetrický (čte se stejně zepředu a zezadu - např. 343). Nalezněte všechny palindromy z intervalu <A,B> pro daná A, B.

153. Nalezněte všechna přirozená čísla menší než dané N, která jsou dělitelná každou svou cifrou.

154. Spočítejte a vytiskněte všechna taková trojciferná čísla, v nichž se žádná cifra neopakuje (tzn. všechny tři cifry čísla jsou různé).

155. Spočítejte a vytiskněte všechna čtyřciferná čísla, která obsahují aspoň tři stejné cifry.

156. Nalezněte všechna přirozená čísla menší než dané N, jejichž dekadický zápis je tvořen pouze ciframi 2, 3 a 5.

157. Nalezněte všechny Pythagorejské trojúhelníky (tj. pravoúhlé trojúhelníky s celočíselnými délkami stran), jejichž všechny tři strany jsou menší než dané N. Hledáme tedy takové trojice celých čísel a,b,c, že $0<a\leq b\leq c<N$ a zároveň $a^2+b^2=c^2$.

158. Vypište všechny různé rozklady daného kladného celého čísla N na součet tří kladných celých čísel. Rozklady lišící se pouze pořadím sčítanců nepovažujeme za různé.

159. Je dáno přirozené číslo N. Jakým nejmenším počtem platidel je možné zaplatit částku N korun, máte-li k dispozici neomezené množství československých mincí a bankovek všech druhů? Vytiskněte nejen celkový počet, ale také rozpis použitých platidel.

160. K danému přirozenému číslu N nalezněte co nejmenší přirozené číslo M takové, že M>N a cifry čísla M jsou permutací cifer čísla N (tzn. číslo M získáme z čísla N změnou pořadí jeho cifer). Příklad: pro N=12542 bude výsledkem M=14225.

161. Je dáno kladné celé K-ciferné číslo, jehož cifry jsou navzájem různé a jsou uspořádány sestupně (klesají). Nalezněte nejbližší větší K-ciferné číslo stejné vlastnosti. Například k danému číslu 86321 byste měli určit číslo 86421.

162. Je dáno kladné celé K-ciferné číslo, jehož cifry se mohou opakovat a jsou uspořádány sestupně (nerostou). Nalezněte nejbližší větší K-ciferné číslo stejné vlastnosti. Například k danému číslu 86321 byste měli určit číslo 86322.

163. Je dána permutace čísel 1 až N, N<10. K dané permutaci nalezněte a vytiskněte nejbližší následující permutaci v lexikografickém uspořádání.

Permutací množiny čísel rozumíme každé její uspořádání. Například pro N=3 vypadají všechny permutace takto:

1 2 3   1 3 2   2 1 3   2 3 1   3 1 2   3 2 1

Lexikografické uspořádání permutací je takové uspořádání, že podíváme-li se na uspořádanou N-tici číslic jako na N-ciferné číslo, jsou tato čísla uspořádána vzestupně podle velikosti. Tedy například výše uvedených 6 permutací čísel 1 až 3 je zapsáno v lexikografickém uspořádání.

164. Pro dané kladné celé číslo N, $N \leq 6$, vytiskněte všechny permutace čísel 1 až N v lexikografickém uspořádání. Pojmy permutace a lexikografické uspořádání jsou vysvětleny v předchozí úloze.

165. Prvočíslo se nazývá permutovatelné, jestliže zůstává prvočíslem, i když zaměníme pořadí jeho cifer (tzn. provedeme permutaci jeho cifer) libovolným způsobem. Například 13 je permutovatelné prvočíslo, protože 13 i 31 jsou prvočísla, obdobně 113 je permutovatelné prvočíslo (neboť 113, 131 i 311 jsou prvočísla). Naopak prvočíslo 103 není permutovatelné, neboť 130 není prvočíslo. Nalezněte a vytiskněte všechna permutovatelná prvočísla menší než 1000.

166. Je dána matice (tj. dvojrozměrné pole) celých čísel velikosti 10x10. Čísla tvořící matici jsou na vstupu zadána po řádcích. Zjistěte, v kolika sloupcích matice je stejný součet všech prvků jako v prvním sloupci.

167. Je dána matice (tj. dvojrozměrné pole) celých čísel velikosti 10x10. Čísla tvořící matici jsou zadána po řádcích. Určete číslo sloupce s maximálním součtem všech prvků. Dosahuje-li se této maximální hodnoty součtu ve více sloupcích, vypište čísla všech takových sloupců.

168. Je dána matice (tj. dvojrozměrné pole) celých čísel velikosti 10x10. Čísla tvořící matici jsou zadána po řádcích. Určete a vypište čísla sloupců, které jsou zcela stejné, jako první sloupec matice (obsahují stejná čísla ve stejném pořadí).

169. Je dána matice (tj. dvojrozměrné pole) kladných celých čísel velikosti 10x10. Čísla tvořící matici jsou zadána po řádcích. Zjistěte, v kolika sloupcích matice jsou obsažena přesně stejná čísla, jako v prvním sloupci (nezáleží na pořadí čísel).

170. Je dána matice (tj. dvojrozměrné pole) kladných celých čísel velikosti 10x10. Čísla tvořící matici jsou zadána po řádcích. Zjistěte, zda je matice symetrická podle hlavní úhlopříčky (tj. úhlopříčky, která vede z levého horního do pravého dolního rohu matice).

171. Je dána matice (tj. dvojrozměrné pole) celých čísel velikosti 10x10. Čísla tvořící matici jsou zadána po řádcích. Přerovnejte čísla v matici tak, aby v každém sloupci byla uspořádána vzestupně podle velikosti. Výslednou matici vytiskněte.

172. Je dána matice (tj. dvojrozměrné pole) celých čísel velikosti 10x10. Čísla tvořící matici jsou zadána po řádcích. Přerovnejte řádky matice tak, aby byly uspořádány vzestupně podle hodnoty svého prvního prvku. Obsah každého řádku matice přitom zůstává zachován, mění se pouze pořadí celých řádků. Výslednou matici vytiskněte.

173. Je dána matice (tj. dvojrozměrné pole) celých čísel velikosti 10x10. Čísla tvořící matici jsou zadána po řádcích. Přerovnejte řádky matice tak, aby byly uspořádány vzestupně podle hodnoty součtu všech svých prvků. Obsah každého řádku matice přitom zůstává zachován, mění se pouze pořadí celých řádků. Výslednou matici vytiskněte.

174. Je dána čtvercová matice celých čísel o rozměrech NxN, N<20. Určete, v kolika řádcích dané matice se nachází permutace čísel 1, 2, ..., N (tzn. každé z čísel 1, 2, ..., N je tam právě jednou).

175. Je dána čtvercová matice celých čísel o rozměrech NxN, N<20. Vypište všechny její řádky, které obsahují permutace čísel 1, 2,..., N (tzn. každé z čísel 1, 2, ..., N je tam právě jednou).

176. Je dána čtvercová matice celých čísel o rozměrech NxN, N<20. Vyměňte v ní největší prvek s nejmenším. Je-li v matici více prvků se stejnou maximální hodnotou nebo naopak více prvků se stejnou minimální hodnotou, můžete použít kterýkoli z nich. Výslednou matici vypište.

177. Je dána čtvercová matice celých čísel o rozměrech NxN, N<20. Váha prvku čtvercové matice NxN je definována jako dvojnásobek jeho hodnoty plus součet hodnot jeho bezprostředních sousedů (těch je maximálně 8). Vyměňte v dané matici prvek s maximální váhou s prvkem s minimální váhou. Výslednou matici vypište.

178. Je dána čtvercová matice celých čísel o rozměrech NxN, N<20. Vyměňte v ní největší prvek nacházející se nad hlavní diagonálou s nejmenším prvkem ležícím pod ní. (Hlavní diagonálou čtvercové matice rozumíme úhlopříčku vedoucí z levého horního do pravého dolního rohu.) Je-li nad hlavní diagonálou více prvků se stejnou maximální hodnotou nebo naopak pod hlavní diagonálou více prvků se stejnou minimální hodnotou, můžete použít kterýkoliv z nich. Výslednou matici vypište.

179. Je dána čtvercová matice celých čísel o rozměrech NxN, N<20. Sedlovým bodem matice nazýváme takový její prvek, který je největším prvkem ve svém řádku a zároveň nejmenším prvkem ve svém sloupci, nebo je nejmenším prvkem ve svém řádku a zároveň největším prvkem ve svém sloupci. Nalezněte sedlový bod dané matice.

Podprogramy

Tento oddíl obsahuje úlohy zaměřené na návrh jednoduchých procedur a funkcí. Většinou se jedná o podprogramy, jejichž tělo je velmi snadné naprogramovat. Cílem je zde procvičit technickou stránku věci, tj. způsob zápisu procedur a funkcí a jejich parametrů.

Při návrhu podprogramů je samozřejmě podstatná správná volba počtu, druhu a způsobu předávání parametrů a vhodné využití navržených podprogramů v hlavním programu. Tyto zásadní úkoly se přirozeným způsobem procvičí až při řešení obtížnějších úloh, například při řešení řady úloh ze III. kapitoly této sbírky.

180. Funkce signum ("znaménko") je funkcí jedné reálné proměnné s tříprvkovým oborem hodnot {-1,0,1}. Je definována tak, že pro kladné vstupní hodnoty dává výsledek 1, pro záporné -1 a pro nulu je výsledkem 0. Naprogramujte funkci signum.

181. Napište funkci se třemi celočíselnými vstupními parametry, která vrací jako svou funkční hodnotu prostřední (tj. druhé největší) ze zadaných tří čísel.

182. Naprogramujte funkci pro výpočet třetí mocniny daného reálného čísla.

183. Naprogramujte funkci pro výpočet n-té mocniny daného reálného čísla, kde n je kladné celé číslo. Parametry funkce budou umocňované číslo a mocnina n.

184. Zapište funkci, která na základě kartézských souřadnic dvou bodů v rovině spočítá jejich vzdálenost. S pomocí této funkce vytvořte program, který ze vstupu přečte souřadnice vrcholů trojúhelníka v rovině a spočítá velikosti všech tří jeho stran.

185. Navrhněte funkci, která zjišťuje, zda je daný znak dekadickou číslicí.

186. Napište funkci, která převádí malá písmena anglické abecedy (tzn. malá písmena bez háčků a bez čárek) na velká. Zadáme-li této funkci jako hodnotu skutečného parametru jiný znak, funkce ho ponechá beze změn.

187. Zapište funkci, která zjišťuje, zda je dané kladné celé číslo prvočíslo. S pomocí této funkce napište program, který čte ze vstupu posloupnost kladných celých čísel ukončenou nulou a vypisuje ta z čísel, která jsou prvočísly.

188. Napište funkci, která počítá v pořadí N-té prvočíslo. Hodnota N se zadává jako parametr funkce.

189. Vytvořte funkci, jejíž jediný vstupní parametr má tvar pole deseti celých čísel. Jako funkční hodnotu bude funkce vracet hodnotu maximálního prvku tohoto pole.

190. Napište funkci na výpočet největšího společného dělitele daných dvou celých čísel. S použitím této funkce vytvořte proceduru, která převede daný zlomek na základní tvar (tj. zkrátí ho co nejvíc, jak je to možné). Zlomek je předáván jako dvojice celých čísel představujících jeho čitatele a jmenovatele.

191. Napište funkci na výpočet nejmenšího společného násobku daných dvou celých čísel.

192. Napište proceduru, která vytiskne řádek obsahující 30 hvězdiček.

193. Napište proceduru, která vytiskne řádek obsahující N znaků Z. Údaje N a Z jsou vstupními parametry procedury.

194. Napište proceduru na převod kartézských souřadnic v rovině na polární. V polárních souřadnicích je bod B v rovině určen svou vzdáleností od počátku soustavy souřadnic P a úhlem, který svírá polopřímka PB s kladnou poloosou souřadnicové osy x.

195. Vytvořte funkci, která zjišťuje, zda je daný rok přestupný. Funkce má pracovat pro hodnoty parametru větší než 1582, tj. pro roky po zavedení nyní používaného Gregoriánského kalendáře. Rok je přestupný tehdy, jestliže je jeho číslo dělitelné čtyřmi. Z tohoto základního pravidla ale existují výjimky v případě celých století. Je-li číslo roku dělitelné stem, je tento rok přestupný pouze

tehdy, pokud je dělitelný také 400. Tedy např. roky 1700, 1800, 1900, 2100 nejsou přestupné, zatímco roky 1600 a 2000 jsou.

196. Navrhněte proceduru, která na základě čísla roku (většího než 1582) a pořadového čísla dne v roce spočítá datum ve tvaru den, měsíc.

197. Ve dvou proměnných X a Y jsou uloženy jisté číselné hodnoty. Pro další práci je potřebujeme buď zachovat beze změny nebo navzájem vyměnit tak, aby v každém případě obsahovaly proměnné X a Y stejnou dvojici čísel jako dřív, a aby navíc platilo X≤Y. Napište proceduru se dvěma parametry, která požadovanou akci provede.

198. Napište proceduru, která z daného pole deseti čísel vybere minimální a maximální hodnotu. Obě tyto hodnoty procedura vrátí jako výsledek.

199. Navrhněte proceduru s jedním parametrem typu pole 20 celých čísel. Procedura v tomto poli nahradí všechny záporné hodnoty nulami, nezáporná čísla ponechá beze změn.

200. Napište proceduru na výměnu řádků v matici. Procedura bude mít tři parametry. Prvním parametrem je matice (tj. dvourozměrné pole) celých čísel velikosti 10x10, dalšími dvěma parametry jsou celá čísla I, J z rozmezí od 1 do 10. Procedura vymění I-tý a J-tý řádek matice.

201. Napište proceduru, která počítá součet dvou čtvercových matic dané velikosti (zvolte si např. velikost 10x10). Matice jsou dvourozměrná pole obsahující jako své prvky libovolná čísla. Matice se sčítají po složkách tak, že hodnota prvku v i-tém řádku a j-tém sloupci výsledné matice je rovna součtu hodnot uložených v i-tém řádku a j-tém sloupci obou sčítaných matic.

Soubory

Úlohy obsažené v tomto oddílu knihy jsou zaměřeny na základní manipulace s datovými a zejména s textovými soubory. Tím se v mnoha případech procvičuje také práce se znaky a znakovými proměnnými.

202. Určete počet znaků tvořících daný textový soubor.

203. Určete počet řádků tvořících daný textový soubor.

204. Vytvořte přesnou kopii daného textového souboru.

205. Jsou dány dva textové soubory. Vytvořte nový textový soubor, který vznikne spojením obsahu obou původních souborů za sebe.

206. Napište program sloužící k rychlému prohlížení obsahu textových souborů. Program z daného textového souboru vypíše prvních 10 řádek (případně méně, je-li soubor kratší).

207. Vytvořte kopii daného textového souboru, ve které na začátek každého řádku vložíte navíc 5 mezer. Tato úprava může být užitečná například pro tisk souboru na tiskárně, pokud chcete posunout levý okraj tisku doprava.

208. Určete počet slov obsažených v daném textovém souboru. Jednotlivá slova jsou od sebe oddělena mezerami (libovolným počtem) a konci řádek.

209. V daném textovém souboru určete počet slov končících písmenem 'A'. Jednotlivá slova jsou od sebe oddělena mezerami a konci řádek.

210. V daném textovém souboru určete počet slov, která začínají a končí stejným písmenem. Jednotlivá slova jsou od sebe oddělena mezerami a konci řádek.

211. V daném textovém souboru určete maximální počet slov na jednom řádku. Jednotlivá slova jsou od sebe oddělena mezerami a konci řádek.

212. Určete délku nejdelšího slova obsaženého v daném textovém souboru. Jednotlivá slova jsou od sebe oddělena mezerami a konci řádek.

213. Nalezněte a vypište nejdelší slovo obsažené v daném textovém souboru. Můžete předpokládat, že jeho délka nebude větší než 30 znaků. V souboru jsou jednotlivá slova od sebe oddělena mezerami a konci řádek.

214. Textový soubor obsahuje dopis. Text dopisu se skládá z vět, každá věta je ukončena tečkou. V jiném významu než pro ukončování vět se znak tečka v dopise nepoužívá. Určete, kolika větami je tvořen dopis. Spočítejte počet znaků (bez mezer mezi jednotlivými slovy) v nejdelší a nejkratší větě.

215. Textový soubor obsahuje dopis. Text dopisu se skládá z vět, každá věta je ukončena tečkou. V jiném významu než pro ukončování vět se znak tečka v dopise nepoužívá. Vytiskněte nejdelší slovo z každé věty. Můžete předpokládat, že všechna slova použitá v dopise jsou kratší než 20 znaků. Jednotlivá slova jsou od sebe oddělena mezerami, konci řádek, čárkami a tečkami ukončujícími věty.

216. Určete počty znaků 'A', 'B' a 'C' v daném textovém souboru.

217. Spočítejte tabulku četností (tj. počty výskytů) všech písmen v daném textovém souboru.

218. Je dán soubor obsahující reálná čísla. Nalezněte největší číslo uložené v tomto souboru.

219. Daný soubor je tvořen celými čísly. Zjistěte, zda je první z čísel větší než poslední.

220. Z daného souboru čísel vytvořte jiný, který bude obsahovat vždy každé třetí číslo z původního souboru.

221. Je dán soubor celých čísel uspořádaný vzestupně podle velikosti, přičemž jednotlivé hodnoty se v souboru mohou

opakovat. Vytvořte nový soubor, který bude obsahovat tatáž čísla, ale již bez opakujících se výskytů.

222. Během v posloupnosti čísel nazýváme maximální souvislý rostoucí úsek posloupnosti (tzn. takový rostoucí úsek, který již nelze prodloužit). Rozdělte danou posloupnost celých čísel z jednoho vstupního souboru do dvou výstupních souborů tak, že liché běhy (tj. v pořadí první, třetí, atd.) se budou ukládat postupně za sebou do jednoho souboru a sudé (tj. druhý, čtvrtý, atd.) do druhého souboru.

223. Jsou dány dva soubory obsahující celá čísla. Čísla v obou souborech jsou uspořádána vzestupně podle velikosti. Vytvořte nový soubor, který bude tvořen čísly z obou původních souborů a bude také uspořádán vzestupně podle velikosti.

224. Jsou dány čtyři soubory obsahující celá čísla. Všechny soubory jsou uspořádány vzestupně podle velikosti. Vytvořte jeden nový soubor, do kterého zapíšete všechna čísla ze všech čtyř vstupních souborů tak, aby byl také uspořádán vzestupně podle velikosti.

225. V daném textovém souboru určete počet výskytů řetězce 'PES'.

226. V daném textovém souboru určete počet výskytů slova 'PES'. Jednotlivá slova jsou od sebe oddělena mezerami a konci řádek.

227. V daném textovém souboru určete počet výskytů řetězce 'PEPES'.

228. Je dán textový soubor. Jednotlivá slova jsou v něm od sebe oddělena mezerami a konci řádek. Vytvořte nový textový soubor, jenž bude tvořen všemi takovými slovy z původního souboru, která se skládají přesně z pěti znaků.

229. Je dán textový soubor. Jednotlivá slova jsou v něm od sebe oddělena mezerami a konci řádek. Vypište z daného souboru všechna slova, která se skládají ze stejných písmen (a ve stejných počtech) jako první slovo souboru.

Dynamické datové struktury

Tento oddíl knihy je orientován na procvičení práce s ukazateli, dynamicky alokovanými proměnnými a se spojovými strukturami. Většina úloh je formulována tak, že úkolem není napsat celý program, ale jen naprogramovat proceduru nebo funkci, která provádí uvedenou operaci s danou dynamickou strukturou.

Úvodní a zároveň nejrozsáhlejší část tohoto oddílu se týká práce s lineárními spojovými seznamy jakožto se základní a nejpoužívanější dynamickou datovou strukturou. Všechny uvedené úlohy předpokládají, že pracujeme s jednosměrným lineárním spojovým seznamem, který má v každém svém prvku jako klíčovou informaci uloženo jedno celé číslo (jen v několika úlohách je to jeden znak). Není-li řečeno jinak, nevylučuje se, že se čísla uložená v jednotlivých prvcích seznamu mohou opakovat, tj. více prvků seznamu může obsahovat stejné číslo. V zadání některých úloh, u nichž je tato skutečnost podstatná pro postup řešení, je to ještě znovu připomenuto. Úlohy je možné modifikovat tak, že změníme množství a druh informace uložené v prvcích seznamu. Zadání všech úloh je také možné změnit tak, že budeme pracovat s obousměrnými spojovými seznamy.

Závěr oddílu obsahuje několik úloh na práci s binárními stromy a s binárními vyhledávacími stromy. Jejich definici zde neuvádíme, k tomu slouží jiné učebnice (doporučujeme například slovenský překlad knihy N. Wirth: Algoritmy a štruktúry údajov, ALFA Bratislava 1988).

230. Napište proceduru, která vytvoří lineární spojový seznam celých čísel. Seznam bude mít 10 prvků s hodnotami po řadě 1, 2,...,10.

231. Napište proceduru, která vytvoří lineární spojový seznam celých čísel. Seznam bude mít 10 prvků s hodnotami, které procedura přečte ze vstupu. Pořadí čísel uložených v seznamu bude opačné, než v jakém byla čísla zadána.

232. Napište proceduru, která vytvoří lineární spojový seznam celých čísel. Seznam bude mít 10 prvků s hodnotami, které procedura přečte ze vstupu. Pořadí čísel uložených v seznamu bude stejné, v jakém byla čísla zadána.

233. V prvcích lineárního spojového seznamu jsou uložena celá čísla. Napište funkci, která určí počet prvků seznamu, obsahujících číslo dělitelné pěti. Např. pro seznam 1, 5, 10, 7, 5 bude výsledkem hodnota 3.

234. Napište proceduru, která z daného lineárního spojového seznamu celých čísel vynechá všechny prvky obsahující číslo dělitelné pěti. Např. ze seznamu 1, 5, 10, 7, 5 tak vznikne upravený seznam 1, 7.

235. V prvcích lineárního spojového seznamu jsou uložena celá čísla. Jednotlivá čísla se mohou v seznamu opakovat. Napište funkci, která určí největší hodnotu čísla uloženého v daném seznamu. Např. pro seznam tvořený prvky s čísly 4, 8, 8, 4, 6, 8, 4 bude výsledkem hodnota 8.

236. V prvcích lineárního spojového seznamu jsou uložena celá čísla. Jednotlivá čísla se mohou v seznamu opakovat. Napište funkci, která určí počet výskytů největší hodnoty čísla v daném seznamu. Např. pro seznam tvořený prvky s čísly 4, 8, 8, 4, 6, 8, 4 bude výsledkem číslo 3, neboť největší hodnota 8 je v seznamu uložena ve 3 prvcích.

237. V prvcích lineárního spojového seznamu jsou uložena celá čísla. Jednotlivá čísla se mohou v seznamu opakovat. Napište proceduru, která z daného seznamu vynechá v pořadí první prvek obsahující největší ze všech čísel v seznamu. Např. ze seznamu 4, 8, 8, 4, 6, 8, 4 tak vznikne upravený seznam 4, 8, 4, 6, 8, 4.

238. V prvcích lineárního spojového seznamu jsou uložena celá čísla. Jednotlivá čísla se mohou v seznamu opakovat. Napište proceduru, která z daného seznamu vynechá poslední prvek obsahující největší ze všech čísel v seznamu. Např. ze seznamu 4, 8, 8, 4, 6, 8, 4 tak vznikne upravený seznam 4, 8, 8, 4, 6, 4.

239. V prvcích lineárního spojového seznamu jsou uložena celá čísla. Jednotlivá čísla se mohou v seznamu opakovat. Napište proceduru, která z daného seznamu vynechá všechny prvky obsahující největší ze všech čísel v seznamu. Např. ze seznamu 4, 8, 8, 4,

240. Napište proceduru, která z daného lineárního spojového seznamu celých čísel vynechá všechny takové prvky, které obsahují stejné číslo jako prvek seznamu bezprostředně předcházející. Např. ze seznamu 1, 4, 4, 4, 5, 4, 9, 9 tak vznikne upravený seznam 1, 4, 5, 4, 9.

241. Napište proceduru, která z daného lineárního spojového seznamu celých čísel vynechá všechny takové prvky, které obsahují stejné číslo jako některý předcházející prvek seznamu. Výsledný seznam tedy bude obsahovat stejná čísla jako původní seznam, ale každé z nich pouze jednou. Např. ze seznamu 1, 4, 4, 4, 5, 4, 9, 9 tak vznikne upravený seznam 1, 4, 5, 9.

242. V prvcích lineárního spojového seznamu jsou uložena celá čísla. Napište proceduru, která v daném lineárním spojovém seznamu všechny prvky zdvojí, tj. za každý prvek vloží jeden nový prvek se stejnou hodnotou. Např. ze seznamu 1, 2, 3, 4 tak vznikne upravený seznam 1, 1, 2, 2, 3, 3, 4, 4.

243. V prvcích lineárního spojového seznamu jsou uložena celá čísla. Napište proceduru, která v daném lineárním spojovém seznamu zdvojí všechny prvky obsahující kladné číslo. To znamená, že za každý prvek s kladným číslem procedura vloží jeden nový prvek se stejnou hodnotou. Např. ze seznamu 1, -2, -3, 4 tak vznikne upravený seznam 1, 1, -2, -3, 4, 4.

244. V prvcích lineárního spojového seznamu jsou uložena celá čísla. Napište proceduru, která daný lineární spojový seznam zdvojí, tj. na konec seznamu připojí ještě kopii každého prvku seznamu se zachováním původního pořadí. Např. ze seznamu 1, 2, 3, 4 tak vznikne upravený seznam 1, 2, 3, 4, 1, 2, 3, 4.

245. Napište proceduru, která přečte ze vstupního textového souboru posloupnost kladných celých čísel ukončenou nulou a vytvoří dva spojové seznamy. Tyto seznamy budou ve svých prvcích obsahovat přečtená kladná čísla (tj. všechna čísla ze vstupu kromě koncové nuly). V prvním seznamu budou uložena všechna přečtená lichá čísla ve stejném pořadí, v jakém byla zadána, ve druhém budou sudá čísla v opačném pořadí, než v jakém byla zadána. Např. pro vstupní posloupnost čísel 11, 28, 24, 17, 99,

66, 60, 0 bude první seznam tvořen prvky s čísly v pořadí 11, 17, 99, druhý seznam bude mít tvar 60, 66, 24, 28.

246. Napište proceduru, která přečte ze vstupního textového souboru zadanou posloupnost znaků a vytvoří dva spojové seznamy znaků. První z nich bude obsahovat všechna velká písmena anglické abecedy (tj. velká písmena bez háčků a čárek) ve stejném pořadí, v jakém byla zadána. Do druhého seznamu se uloží všechny číslice, a to v opačném pořadí, než v jakém byly zadány. Např. pro zadanou posloupnost znaků 7z6A**35BC bude první seznam obsahovat po řadě prvky se znaky A, B, C, zatímco druhý seznam bude mít tvar 5, 3, 6, 7.

247. Napište funkci typu boolean, která zjišťuje, zda jsou dané dva lineární spojové seznamy celých čísel stejné, tzn. zda jsou tvořeny stejnými prvky ve stejném pořadí.

248. Napište funkci typu boolean, která zjišťuje, zda jsou dané dva lineární spojové seznamy celých čísel (obecně neuspořádané) tvořeny stejnými prvky. Pro jednoduchost předpokládejte, že v žádném ze seznamů se čísla neopakují. Po ukončení funkce, dané seznamy čísel nemusejí zůstat zachovány. Např. pro seznamy 5, 7, 9, 11 a 11, 5, 9, 7 bude správná odpověď znít true.

249. Napište proceduru, která daný spojový seznam kladných celých čísel rozdělí na K seznamů tak, že do každého z výsledných seznamů zařadí vždy všechna čísla dávající stejný zbytek při celočíselném dělení číslem K. Hodnota K je zadána parametrem, předem je znám její horní odhad (například 10). Např. pro vstupní seznam 1, 7, 5, 14, 2, 9, 6, 16, 4, 19, 8 a hodnotu K=3 vytvoří procedura následující tři seznamy: první obsahuje čísla dělitelná třemi a má tvar 9, 6, druhý seznam je tvořen čísly 1, 7, 16, 4, 19 (dávají při dělení třemi zbytek 1) a třetí vytvořený seznam obsahuje zbývající čísla 5, 14, 2, 8, která při dělení třemi dají zbytek 2.

250. Dva lineární spojové seznamy obsahují ve svých prvcích celá čísla. V prvním ze zadaných seznamů jsou všechna čísla navzájem různá. Napište proceduru, která v prvním seznamu najde prvek s největším číslem a tento prvek přeřadí na konec

druhého seznamu. Např. pro vstupní seznamy s čísly 11, 70, 35 a 25, 99, 6, 18 vede požadovaná úprava k seznamům ve tvaru 11, 35 a 25, 99, 6, 18, 70.

251. Dva lineární spojové seznamy obsahují ve svých prvcích celá čísla. V prvním ze zadaných seznamů jsou všechna čísla navzá- jem různá. Napište proceduru, která v prvním seznamu najde prvek s největším číslem. Tento nalezený prvek potom procedura přeřadí na konec druhého seznamu, pokud ve druhém seznamu dosud není žádný prvek s tímto číslem. V opačném případě se nalezený prvek z prvního seznamu vypustí a zruší. Např. pro vstupní seznamy s čísly 11, 70, 35 a 25, 99, 6, 18 vede požado- vaná úprava k seznamům ve tvaru 11, 35 a 25, 99, 6, 18, 70. Naproti tomu pro seznamy 11, 70, 35 a 25, 70, 6, 18 bude mít po úpravě první seznam rovněž tvar 11, 35, ale druhý seznam zůstane beze změn (číslo 70 v něm již bylo).

252. Dva lineární spojové seznamy obsahují ve svých prvcích celá čísla. Oba seznamy jsou uspořádané vzestupně podle velikosti uložených čísel. Napište proceduru, která přeřadí prvek s největ- ším číslem (tj. poslední prvek) z prvního seznamu do druhého seznamu tak, aby druhý seznam zůstal uspořádaný. Např. pro vstupní seznamy 11, 35, 70 a 22, 25, 91, 99 budou mít výsledné seznamy tvar 11, 35 a 22, 25, 70, 91, 99.

253. Dva lineární spojové seznamy obsahující ve svých prvcích celá čísla jsou uspořádány vzestupně podle velikosti. Napište proce- duru, která přesune všechny jejich prvky do jednoho uspořáda- ného seznamu. Např. pro vstupní seznamy 11, 35, 70 a 22, 25, 91, 99 bude mít výsledný seznam tvar 11, 22, 25, 35, 70, 91, 99.

254. Napište proceduru, která obrací pořadí prvků v jednosměrném lineárním spojovém seznamu. Např. ze seznamu 2, 4, 6, 8, 10 tak vznikne seznam obsahující prvky v pořadí 10, 8, 6, 4, 2.

255. Množiny jsou reprezentovány jednosměrnými spojovými seznamy svých prvků (každý prvek množiny je v seznamu uložen právě jednou). Napište proceduru, která v této reprezentaci provádí operaci sjednocení dvou množin. Např. pro dva vstupní sez- namy, z nichž jeden obsahuje (v libovolném pořadí) prvky 1, 3,

5, 6, 8 a druhý 2, 3, 4, 5, 6, bude výsledný seznam obsahovat (opět v libovolném pořadí) prvky 1, 2, 3, 4, 5, 6, 8.

256. Množiny jsou reprezentovány jednosměrnými spojovými seznamy svých prvků (každý prvek množiny je v seznamu uložen právě jednou). Napište proceduru, která v této reprezentaci provádí operaci průnik množin. Např. pro dva vstupní seznamy, z nichž jeden obsahuje (v libovolném pořadí) prvky 1, 3, 5, 6, 8 a druhý 2, 3, 4, 5, 6, bude výsledný seznam obsahovat (opět v libovol- ném pořadí) prvky 3, 5, 6.

257. Řešte předchozí dvě úlohy jestliže je množina reprezentována uspořádaným lineárním spojovým seznamem svých prvků.

258. Velké přirozené číslo je v programu reprezentováno jako lineárním spojovým seznamem svých cifer. Napište proceduru nebo funkci pro sčítání takto reprezentovaných čísel.

259. Velké přirozené číslo je v programu reprezentováno lineárním spojovým seznamem cifer. Napište proceduru nebo funkci pro porovnávání takto reprezentovaných čísel. Výsledkem práce procedury (funkce) bude informace, zda je první ze zadaných čísel menší nebo větší než druhé číslo, případně zda se obě čísla sobě rovnají.

260. Napište proceduru, která přečte ze vstupního textového souboru dvě velká kladná celá čísla (každé je zapsáno na jednom řádku souboru) a vytiskne jejich součet. Velikost čísel není omezena.

261. Napište proceduru, která přečte ze vstupního textového souboru tři velká kladná celá čísla (každé je zapsáno na jednom řádku souboru) a vytiskne jejich součet. Velikost čísel není omezena.

262. Napište funkci na přesný výpočet hodnoty N! (faktoriál) pro dané kladné celé číslo N. Výslednou hodnotu předejte ve vhodné reprezentaci pomocí lineárního spojového seznamu.

263. Napište funkci na přesný výpočet hodnoty 2^N pro dané kladné celé číslo N. Výslednou hodnotu předejte ve vhodné reprezentaci pomocí lineárního spojového seznamu.

264. Napište proceduru, která vytvoří lineární spojový seznam celých čísel. Seznam bude mít 10 prvků s hodnotami, které procedura přečte ze vstupu. Ve vytvořeném seznamu budou čísla uspořádána vzestupně podle velikosti.

265. Je dán lineární spojový seznam obsahující ve svých prvcích celá čísla. Napište proceduru, která prvky seznamu uspořádá vzestupně podle hodnot uložených čísel. Není povoleno měnit celočíselné hodnoty uložené v jednotlivých prvcích seznamu, smí se pouze přepojovat celé prvky seznamu.

266. Znakové řetězce jsou reprezentovány jednosměrnými spojovými seznamy znaků. Napište proceduru, která v řetězci A nahradí první výskyt řetězce B řetězcem C (kde A, B, C jsou parametry procedury).

267. Znakové řetězce jsou reprezentovány jednosměrnými spojovými seznamy znaků. Napište proceduru, která z řetězce A vynechá poslední výskyt řetězce B (kde A, B jsou parametry procedury).

268. Znakové řetězce jsou reprezentovány jednosměrnými spojovými seznamy znaků. Napište proceduru, která daný řetězec doplní na dvojnásobnou délku zrcadlovým obrazem původního řetězce.

269. Znakové řetězce jsou reprezentovány jednosměrnými spojovými seznamy znaků. Napište funkci, jejíž funkční hodnotou je počet výskytů řetězce A v řetězci B. Výskyty řetězce A v řetězci B se mohou částečně překrývat. A a B jsou parametry funkce.

270. Polynom je reprezentován pomocí lineárního spojového seznamu tak, že v každém prvku seznamu jsou uloženy hodnoty koeficientu a exponentu jednoho členu polynomu. Prvky seznamu jsou uspořádány sestupně podle hodnot exponentu. Členy s nulovým koeficientem se do seznamu neukládají. Napište proceduru, která přečte ze vstupu údaje o koeficientech a exponentech daného polynomu a vytvoří příslušnou datovou reprezentaci polynomu.

271. Polynom je reprezentován pomocí lineárního spojového seznamu tak, že v každém prvku seznamu jsou uloženy hodnoty koeficientu a exponentu jednoho členu polynomu. Prvky seznamu jsou uspořádány sestupně podle hodnot exponentu. Členy s nulovým koeficientem se do seznamu neukládají. Napište proceduru, která na základě uvedené reprezentace polynomu tento polynom vytiskne.

272. Polynom je reprezentován pomocí lineárního spojového seznamu tak, že v každém prvku seznamu jsou uloženy hodnoty koeficientu a exponentu jednoho členu polynomu. Prvky seznamu jsou uspořádány sestupně podle hodnot exponentu. Členy s nulovým koeficientem se do seznamu neukládají. Napište proceduru na sčítání polynomů.

273. Polynom je reprezentován pomocí lineárního spojového seznamu tak, že v každém prvku seznamu jsou uloženy hodnoty koeficientu a exponentu jednoho členu polynomu. Prvky seznamu jsou uspořádány sestupně podle hodnot exponentu. Členy s nulovým koeficientem se do seznamu neukládají. Napište proceduru na násobení polynomů.

274. Je dán binární strom, který má v každém svém uzlu uloženo jedno celé číslo. Napište proceduru, která projde daným binárním stromem do šířky (tj. po vrstvách stromu) a při průchodu každým uzlem vytiskne hodnotu uloženého čísla.

275. Napište proceduru, která přečte ze vstupního textového souboru posloupnost kladných celých čísel ukončenou nulou a vytvoří binární vyhledávací strom, do kterého uloží všechna přečtená čísla.

276. Napište proceduru, která přečte ze vstupního textového souboru posloupnost (alespoň tří) kladných celých čísel ukončenou nulou a vytvoří binární vyhledávací strom, do kterého uloží všechna přečtená čísla kromě dvou největších z nich.

277. V binárním vyhledávacím stromu jsou uložena celá čísla. Napište proceduru, která z něho vypustí uzel s daným číslem, pokud tam je takové číslo vůbec uloženo. Není-li dané číslo ve stromě obsaženo, procedura ponechá binární vyhledávací strom beze změn. Parametry procedury jsou odkaz na kořen stromu a vypouštěné číslo.

278. V binárním vyhledávacím stromu jsou uložena celá čísla. Napište proceduru, která z něho vypustí všechny uzly s hodnotami vyššími, než je dané číslo C. Parametry procedury jsou odkaz na kořen stromu a číslo C.

279. V uzlech binárního vyhledávacího stromu jsou uložena celá čísla. Napište proceduru, která vypíše ohodnocení všech uzlů daného binárního vyhledávacího stromu v pořadí podle rostoucích hodnot klíčů.

280. V uzlech binárního vyhledávacího stromu jsou uložena celá čísla. Napište proceduru, která vypíše ohodnocení všech uzlů daného binárního vyhledávacího stromu v pořadí podle rostoucích hodnot klíčů. Nepoužívejte rekurzi.

281. Napište proceduru, která vytvoří vyvážený binární vyhledávací strom o N vrcholech (N je parametr), v němž budou vrcholy ohodnoceny čísly 1 až N. Vyvážený strom je takový, ve kterém se počet vrcholů v levém a pravém podstromu libovolného vrcholu liší nejvýše o jednu.

282. Množinu budeme reprezentovat binárním vyhledávacím stromem, ve kterém jsou uloženy právě všechny prvky množiny. Napište proceduru, která v této prezentaci provádí operaci sjednocení množin.

283. Množinu budeme reprezentovat binárním vyhledávacím stromem, ve kterém jsou uloženy právě všechny prvky množiny. Napište proceduru, která v této reprezentaci provádí operaci průnik množin.

284. Navrhněte programovou realizaci obecného stromu (tzn. není omezen počet synů jednoho uzlu) pomocí dynamických datových struktur. V každém uzlu stromu bude jako informační složka uloženo jedno celé číslo. Napište funkci typu boolean, která zjišťuje, zda se v daném stromě nachází dané číslo (odkaz na kořen stromu a hledané číslo jsou parametry funkce).

Obtížnější úlohy

Výpočty s čísly

285. Trojúhelník je zadán třemi údaji, z nichž jedním je vždy velikost jedné jeho strany a zbývající dva údaje jsou velikosti dalších stran nebo úhlů. Ve vstupních datech je vhodným způsobem vyznačeno, který ze vstupních údajů je délkou strany a který je velikostí úhlu. Vypočtěte chybějící délky stran a velikosti úhlů v trojúhelníku.

286. Dva obdélníky nazýváme nezávislé, jestliže není možné vložit jeden do druhého. Vypište rozměry všech dvojic nezávislých obdélníků takových, které mají obě strany celočíselné délky menší než 20.

287. Řešte soustavu tří lineárních rovnic o třech neznámých. Soustava rovnic je zadána seznamem koeficientů u jednotlivých členů rovnic ve vhodném pořadí, jaké si zvolíte.

288. Řešte soustavu N lineárních rovnic o N neznámých. Hodnota N určující velikost soustavy je dána. Můžete předpokládat, že N nepřekročí hodnotu 10. Soustava rovnic je pak na vstupu zadána seznamem koeficientů u jednotlivých členů rovnic ve vhodném pořadí.

289. Napište program pro počítání velkých mocnin matice. Prvky matice jsou celá čísla. Vstupem programu jsou hodnoty prvků matice (po řádcích) a celé kladné číslo udávající mocninu, kterou máme spočítat.

290. Je dáno celé kladné číslo N, $N<50$. Spočítejte přesně hodnotu N! (N faktoriál). Číslo N! je definováno jako součin všech celých kladných čísel od 1 do N, tj. $N!=1\cdot 2\cdot \cdots \cdot N$.

291. Je dáno přirozené číslo N menší než 100. Spočítejte přesnou hodnotu 2^N (dvě na N).

292. Jsou dána dvě "dlouhá" kladná celá čísla, tj. čísla s velkým počtem cifer (větším, než je povolený rozsah hodnot celočíselných proměnných v Pascalu). Jako vstupní data programu jsou obě čísla uvedena bez znaménka, počet cifer žádného z nich není větší než 100. Spočítejte a vytiskněte jejich součet.

293. Jsou dána dvě "dlouhá" kladná celá čísla, tj. čísla s velkým počtem cifer (větším, než je povolený rozsah hodnot celočíselných proměnných v Pascalu). Na vstupu programu jsou uvedena bez znaménka, počet jejich cifer není předem znám. Spočítejte a vytiskněte jejich součin.

294. Napište proceduru, která spočítá celočíselný podíl a zbytek po celočíselném dělení dvou velkých celých kladných čísel. Počet cifer obou čísel není předem znám (může být větší, než je přípustný rozsah hodnot celočíselných proměnných v Pascalu).

295. Jsou dána dvě kladná celá čísla. Spočítejte přesně jejich podíl. Výsledkem bude reálné číslo určené na tolik desetinných míst, že buď další desetinná místa obsahují samé nuly, nebo se v desetinném rozvoji objevila perioda (v takovém případě určete její délku).

296. Jsou dány dva polynomy jedné proměnné. Každý polynom je zadán seznamem dvojic čísel představujících vždy koeficienta exponent jednotlivých členů polynomu. Spočítejte podíl těchto polynomů a zbytkový polynom.

297. Najděte všechny alespoň trojčlenné aritmetické posloupnosti prvočísel menších než dané kladné celé číslo N. Vypisujte jen maximální posloupnosti (v tom smyslu, že je není možné prodloužit).

298. Je dána posloupnost 100 celých čísel. Zjistěte délku co možná nejdelší rostoucí podposloupnosti, kterou je možné z dané posloupnosti vybrat. Vybraná podposloupnost je tvořena některými prvky zadané posloupnosti se zachováním jejich původního vzájemného pořadí, přičemž tyto prvky v dané posloupnosti nemusí tvořit souvislý úsek. Například nejdelší rostoucí podposloupností vybranou z posloupnosti čísel 10 4 16 6 12 9 12 7 13 je posloupnost 4 6 9 12 13, která má délku 5.

299. Všechna existující taková čtyřciferná čísla, která nemají ve svém dekadickém zápisu dvě stejné cifry, jsou seřazena do posloupnosti podle velikosti. Spočítejte pro danou hodnotu K v pořadí K-tý člen této posloupnosti.

300. Přirozená čísla, jejichž dekadický zápis se skládá pouze z číslic 2, 3 a 5, jsou seřazena do posloupnosti podle velikosti. Spočítejte pro danou hodnotu K v pořadí K-tý člen této posloupnosti.

301. Napište program, který k zadanému celému číslu N spočítá a vytiskne prvních N kladných celých čísel takových, která se dají vyjádřit ve tvaru součinu $3^i \cdot 4^j \cdot 5^k$, kde i, j, k jsou nezáporná celá čísla. Například pro N=10 program vytiskne následujících 10 čísel: 1 3 4 5 9 12 15 16 20 25.

302. Je dána posloupnost kladných celých čísel. Napište program, který zjistí, zda je tato posloupnost součtová. Posloupnost kladných celých čísel nazveme součtovou, jestliže začíná číslem 1 a každý další její člen lze vyjádřit jako součet nějakých dvou čísel (různých nebo stejných), která se v posloupnosti již dříve vyskytla. Například posloupnost 1, 2, 4, 5, 3 je součtová, neboť 2=1+1, 4=2+2, 5=1+4 a 3=1+2.

303. V předchozí úloze byl zaveden pojem součtová posloupnost. Je dána posloupnost kladných celých čísel. Napište program, který zjistí, zda lze nějakou změnou pořadí jejich prvků získat součtovou posloupnost. Například přerovnáním posloupnosti 3, 5, 1, 3, 2, 4, 3 lze získat součtovou posloupnost 1, 2, 3, 5, 3, 4, 3.

304. Napište program, který k zadané posloupnosti kladných celých čísel vytiskne rostoucí posloupnost (navzájem různých!) všech takových čísel, která vzniknou sečtením některých dvou čísel obsažených v zadané posloupnosti. Například pro posloupnost 2, 11, 1, 3, 2 program vypíše posloupnost 2, 3, 4, 5, 6, 12, 13, 14, 22, neboť 2=1+1, 3=1+2, 4=2+2, 5=2+3, 6=3+3, 12=1+11, 13=2+11, 14=3+11, 22=11+11. Předpokládejte, že zadaná posloupnost se vejde do paměti, ale výsledná se již do paměti vejít nemusí (může obsahovat více prvků než posloupnost původní).

305. Pro dané přirozené číslo N rozdělte čísla 1, 2, 3,..., N^2 do N skupin tak, aby v každé skupině bylo přesně N čísel, a aby všechny skupiny měly stejný součet.

306. Napište program na vyhodnocení aritmetického výrazu tvořeného celočíselnými konstantami, běžnými operátory +, -, *, / a závorkami s libovolným počtem úrovní vnoření. Výraz vyhodnocujte v oboru celých čísel, binární operátor / zde představuje celočíselné dělení. Při vyhodnocování výrazu uvažujte obvyklou prioritu operátorů (tj. násobení a dělení mají stejnou prioritu, a to vyšší než sčítání a odčítání, operátory téže priority následující bezprostředně po sobě se vyhodnocují zleva doprava).

307. Postfixová notace je takový způsob zápisu aritmetických výrazů, ve kterém operátor vždy bezprostředně následuje za těmi operandy, k nimž se vztahuje. Každým z těchto operandů může být číselná konstanta nebo opět výraz. Tak například výraz 3+4 zapíšeme v postfixové notaci jako 3 4 +, výraz 5*(6+7) bude mít tvar 5 6 7 + *, výraz (5-2)+(4-3) má postfixový zápis 5 2 - 4 3 - +. Všimněte si, že v zápisu výrazu v postfixové notaci je zachováno stejné pořadí číselných konstant jako v obvyklém (tzv. infixovém) zápisu, a dále že není zapotřebí používat závorky. Správného pořadí provádění dílčích operací se zde dosáhne vhodným umístěním operátorů ve výrazu. Například (3+4)*5 se zapíše 3 4 + 5 *, zatímco 3+(4*5) zapíšeme jako 3 4 5 * +. Na vstupu je zadán aritmetický výraz v postfixové notaci. Výraz je tvořen celočíselnými konstantami bez znaménka a dále běžnými binárními aritmetickými operátory +, -, *, /. Jednotlivé číselné konstanty a znaménka jsou na vstupu odděleny mezerami. Napište program na vyhodnocení zadaného výrazu. Výraz vyhodnocujte v oboru celých čísel, binární operátor / zde představuje celočíselné dělení.

308. Je zadán aritmetický výraz tvořený celočíselnými konstantami, běžnými operátory +, -, *, / a závorkami s libovolným počtem úrovní vnoření. Převeďte tento výraz do postfixové notace. Postfixová notace aritmetických výrazů je popsána v zadání předchozí úlohy.

309. Nalezněte všechny pětice kladných celých čísel x_1, x_2, x_3, x_4, x_5 takové, pro které platí:
$x_1 \le x_2 \le x_3 \le x_4 \le x_5$ a $x_1 + x_2 + x_3 + x_4 + x_5 = N$,
kde N je zadané kladné celé číslo.

310. Je dáno přirozené číslo N, $N \le 100$. Vypište všechny možné výplaty N korun pomocí platidel s hodnotami 1, 2, 5, 10, 20 a 50 korun.

311. Napište program, který vypíše všechny rozklady daného kladného celého čísla N na součet kladných celých čísel. Rozklady, které se liší pouze pořadím sčítanců, považujeme za stejné. Např. číslo 5 je možné rozložit šesti způsoby:
5=4+1=3+2=3+1+1=2+2+1=2+1+1+1=1+1+1+1+1

312. Lístek do kina stojí 10 Kčs. Ve frontě stojí 2N lidí, přesně N má desetikorunu a N dvacetikorunu. Každý člověk ve frontě chce koupit jeden lístek. V pokladně kina nejsou na začátku prodeje žádné peníze. Napište program, který pro dané N vypíše všechna taková rozmístění lidí ve frontě, pro která bude mít pokladní vždycky nazpátek. Určete počet všech takových rozmístění.

313. Nalezněte a vytiskněte všechny latinské čtverce řádu 4. Latinský čtverec řádu 4 je matice velikosti 4x4, která obsahuje pouze čísla 1 až 4, a to tak, že v každém jejím řádku i sloupci jsou všechny prvky různé.

314. Nalezněte a vytiskněte všechny diagonální latinské čtverce řádu 4. Diagonální latinský čtverec řádu 4 je matice velikosti 4x4, která obsahuje pouze čísla 1 až 4, a to tak, že v každém jejím řádku, sloupci i na obou úhlopříčkách jsou všechny prvky různé.

315. Nalezněte a vytiskněte alespoň jeden magický čtverec řádu N. Magický čtverec řádu N je matice velikosti NxN, která obsahuje právě jednou každé z čísel od 1 do N^2. Přitom platí, že součet čísel v každém řádku, v každém sloupci i na obou úhlopříčkách je stejný. Předpokládejte vhodné omezení na přípustné hodnoty velikosti čtverce N (např. N<8).

Kódování a úpravy textů

316. Napište procedury, které budou převádět kladná celá čísla do tvaru římských čísel a zpět.

317. Napište procedury nebo funkce, které budou převádět daný text do morseovky a zpět. Stačí, když budete umět kódovat písmena a číslice.

318. Napište podprogramy, které budou umožňovat vstup a výstup celých čísel v jiné soustavě než desítkové. Rozsah přípustného základu číselné soustavy můžete vhodně omezit. Procedura pro výstup bude umožňovat také formátování výstupu podobně, jako ho známe z Pascalu (tisk na daný počet míst).

319. Napište program, který bude vypisovat zadanou číselnou hodnotu slovně (jako na složenku). Omezte se na kladná celá čísla menší než miliarda.

320. Napište program pro převádění vzorců anorganické chemie do správného českého názvosloví. Vstupem programu je několik vzorců ve vhodně zvoleném tvaru zápisu, výstupem jsou názvy odpovídajících chemických sloučenin zapsané slovně. Předem vymezte typy sloučenin, s nimiž bude program umět pracovat (např. soli anorganických kyselin).

321. Napište program na sestavování a vykreslování strukturních vzorců základních organických sloučenin. Předem vymezte, se kterými typy sloučenin bude program pracovat (např. alkany, alkeny, alkiny, alkadieny, ..., aldehydy, alkoholy, halogenderiváty, uhlovodíky s deriváty odvozenými od alkanů, apod.) a vhodně si omezte velikost sloučenin (např. počet uhlíkových atomů v hlavním řetězci nebude větší než 10).

322. Je dán textový soubor a přirozené číslo K. Zašifrujte daný text tak, že každé velké písmeno nahradíte písmenem, které je v abecedě o K znaků dále. Tento posun v abecedě uvažujte cyklicky (tzn. po 'Z' následuje 'A' atd.). Ostatní znaky obsažené v daném textu přeneste do výsledného souboru beze změn.

323. Napište program na kódování a dekódování textových souborů tzv. Césarovou šifrou. Text se šifruje posunem písmen v abecedě. Na vstupu programu je zadán klíč šifry. Je to slovo tvořené velkými písmeny anglické abecedy, v němž A značí posun o 1 písmeno, B o dvě písmena, atd., až Z představuje posun v abecedě o 26 písmen. Posun šifrovaného znaku v abecedě se přitom chápe cyklicky, tj. po písmenu Z následuje opět A. Pro jednoduchost můžete předpokládat, že šifrovaný text je tvořen pouze velkými písmeny anglické abecedy. Je-li šifrovaná zpráva delší než klíč, rozdělí se celá zpráva na úseky stejné délky, jakou má klíč, a stejný klíč se pak použije opakovaně pro zašifrování každého úseku. Např. slovo "KOLIK" zakódované podle klíče "AB" je "LQMKL".

324. Napište program na kódování a dekódování textových souborů pomocí šifrovací mřížky. Text se nejprve zapíše po jednotlivých znacích do pomocné čtvercové matice. Klíčem je čtvercová mřížka velikosti NxN, kde N je malé sudé číslo. V mřížce je vyřezáno přesně $N^2/4$ otvorů tak, aby při postupném otáčení mřížky o 90 stupňů nikdy nepřišly dva různé otvory na stejné místo. Šifrovaný text se vpisuje po znacích do otvorů v mřížce postupně po řádcích. Po zapsání $N^2/4$ znaků (tj. do každého otvoru mřížky jeden znak) se mřížka otočí o 90° a další text se opět vpisuje do otvorů. Stejně postupujeme i pro další polohy mřížky, dokud nezapíšeme do matice N^2 znaků textu. Poté mřížku "odstraníme" a zaplněnou matici znaků vypíšeme po řádcích. Tím je text zašifrován. Je-li šifrovaná zpráva delší než N^2 znaků (tj. jeden zcela zaplněný čtverec), rozdělí se na více kratších úseků délky N^2, případně poslední úsek je kratší a zbylá volná místa v matici se pak vyplní mezerami. Vstupem šifrovacího programu je údaj o velikosti mřížky N, obsah šifrovací mřížky a textový soubor obsahující šifrovaný text. Výsledkem práce bude textový soubor se zašifrovaným textem.

325. Napište program na formátování textů. Vstupem programu je textový soubor. Slovem v textu nazveme takový úsek textu, který je z obou stran oddělen jednou nebo více mezerami, či konci řádků. Napište program, který zadaný text vytiskne se zarovnanými okraji na levé a pravé straně. Dosáhnete toho tím, že mezi slova umístěná na jednom řádku vložíte vhodný počet mezer. Šířka výsledného tisku se zadává na vstupu programu.

326. Napište program sloužící ke zformátování zadaného textu do požadovaného tvaru. Pokyny k formátování jsou zadávány jako zvláštní příkazy přímo v textu a při formátování jsou z textu vynechávány. Od textu se formátovací příkazy odliší tím, že jsou umístěny na samostatných řádcích, které začínají speciálním vyhrazeným znakem, např. '#'. Je možné stanovit šířku tiskového sloupce a polohu jeho levého okraje a průběžně je měnit. Dále je možné předepsat umístění odstavců, vložení volných řádek do textu a odstránkování. Příkazy umožňují požadovat v rámci odstavce zarovnání textu k pravému okraji rozšířením mezer mezi slovy textu. Musí být ale také zachována možnost ponechat část vstupního textu v nezměněném tvaru.

327. Napište program sloužící k tisku plakátů. Program přečte ze vstupu zadaný nápis (řetězec znaků) a vytiskněte ho na tiskárně velkými písmeny (např. v rastru 5×7 bodů nebo větším). Zvolte vodorovný nebo svislý směr tisku na papíře podle délky nápisu.

328. Napište program sloužící k vytváření a tisku jízdních řádů pro jednotlivé linky tramvají a autobusů. Na vstupu je pro zvolenou linku popsána trasa (tzn. počet zastávek, jejich názvy a doba jízdy mezi jednotlivými dvojicemi po sobě následujících zastávek) a jsou uvedeny požadované intervaly mezi jednotlivými spoji v různou denní dobu. Výsledkem programu je výpis jízdních řádů pro řidiče jednotlivých spojů a vytištění jízdních řádů na jednotlivé zastávky.

329. Napište program, který vytiskne zadaný aritmetický výraz v přehledném tvaru, tj. s vodorovnými zlomkovými čárami včetně možnosti složených zlomků. V případě složených zlomků je třeba odlišit různou délkou zlomkové čáry pořadí vyhodnocování. Vstupem programu bude zápis výrazu, který se má vytisknout, ve tvaru obvyklém v programovacích jazycích.

330. Napište program sloužící k úpravám textových souborů. Program bude mít dva vstupy: prvním vstupem bude zpracovávaný textový soubor a druhým pomocný textový soubor obsahující editační příkazy. Editační příkazy se odvolávají na jednotlivé řádky zpracovávaného souboru pomocí jejich pořadových čísel a umožňují např. vypuštění řádku, náhradu řádku jiným, náhradu jednoho řetězce znaků jiným řetězcem znaků, vypouštění a přemísťování bloku textu (tj. několika po sobě jdoucích řádků) atd. Přesný tvar editačních příkazů si zvolte sami. Výsledný text po provedení požadované editace uložte do výstupního souboru.

331. Napište program sloužící k úpravě textu programu zapsaného v programovacím jazyce Pascal. Úpravou se zde rozumí vhodné zformátování textu tak, aby byl zapsaný program dobře čitelný (např. rozdělení jednotlivých příkazů na samostatné řádky, odsazování začátků řádků od levého okraje stránky podle vnoření řídicích struktur, vložení prázdných řádků k oddělení jednotlivých úseků deklarací apod.). Formát upraveného textu může být buď zvolen pevně, nebo bude možné ovlivňovat ho pomocí parametrů zadávaných ze vstupu (o kolik odsazovat, šířka řádku, možnost číslování řádků atd.).

332. Napište program na výpis tzv. křížových referencí v Pascalu. Vstupem programu bude textový soubor obsahující nějaký syntakticky správný program zapsaný v programovacím jazyce Pascal. Výsledkem práce programu je jiný textový soubor, v němž je uložen nejprve tentýž text programu, ale navíc s očíslovanými řádky, a dále tabulka křížových referencí. Tato tabulka obsahuje jména všech v programu použitých identifikátorů s uvedením čísel řádků, kde se v programu vyskytují.

333. V předchozí úloze je popsán program na výpis křížových referencí v Pascalu. Řešte stejnou úlohu, ale ve výsledné tabulce u každého identifikátoru navíc zvlášť vyznačte, který výskyt identifikátoru je tzv. definiční (identifikátor je tam deklarován). U identifikátorů proměnných navíc rozlišujte, kde je identifikátor v programu použit, aniž by došlo ke změně hodnoty příslušné proměnné, a kde se hodnota proměnné může změnit (dosazením nebo čtením hodnoty ze vstupu, případně u řídicí proměnné ve for-cyklu).

334. Řešte stejnou úlohu na sestavení tabulky křížových referencí v Pascalu, jako je popsána v předchozích dvou úlohách. Navíc však rozlišujte lokalitu proměnných. Stejný identifikátor může být v programu použit pro označení dvou nebo i více různých objektů (např. proměnných, konstant, procedur apod.), pokud se

deklarace každého z nich nachází v jiném bloku (podprogramu). Ve vytvářené tabulce křížových referencí takováto použití téhož identifikátoru odlište.

335. Řešte stejnou úlohu na sestavení tabulky křížových referencí v Pascalu, jako je popsána v předchozích úlohách. Výslednou tabulku křížových referencí uspořádejte po jednotlivých podprogramech (hlavní program, procedury, funkce). Vždy ke každému jménu procedury nebo funkce bude v tabulce uveden seznam všech identifikátorů, které se v ní vyskytují a čísla řádků jednotlivých výskytů. Některé (globální) proměnné nemusí mít v proceduře žádný definiční výskyt.

336. V syntakticky správném programu zapsaném v jazyce Pascal doplňte ke každému klíčovému slovu begin a end pomocný komentář, který bude tato klíčová slova blíže identifikovat, a tím usnadní orientaci v programu. Komentář bude obsahovat tyto informace: jednoznačné identifikační číslo shodné vždy pro begin a jemu odpovídající end, úroveň vnoření, případně jméno procedury nebo funkce, jejíž tělo tento begin zahajuje nebo end ukončuje.

Zpracování dat

337. Zpracujte výsledky jednoho závodu dvojkombinace světového poháru v lyžování. Dvojkombinace se skládá ze sjezdu a obřího slalomu. Vstupem programu je seznam jmen závodníků s přidělenými startovními čísly, a dále seznamy výsledných časů dosažených zvlášť ve sjezdu a zvlášť v obřím slalomu. Tyto seznamy jsou uspořádány podle startovních čísel. Vytiskněte přehledné výsledkové listiny sjezdu a obřího slalomu a výsledné pořadí závodníků v dvojkombinaci. To je dáno součtem umístění v obou dílčích závodech.

338. Zpracujte výsledné pořadí jednoho ročníku světového poháru v lyžování. Vstupem programu jsou výsledkové listiny jednotlivých závodů poháru zadané ve vhodném tvaru. U každé listiny je poznamenáno, zda se jedná o sjezd, superobří slalom, obří slalom nebo slalom (kombinace zde pro jednoduchost neuvažujeme). Vytiskněte pořadí závodníků v jednotlivých disciplínách a celkové pořadí světového poháru. Pořadí se stanoví tak, že závodníci dostávají body za umístění v každém závodě: za vítězství je 25 bodů, za 2. místo 20 bodů, za 3. místo 15 bodů, za 4. místo 12 bodů, a dále vždy o bod méně, až za 15. místo je 1 bod.

339. Sestavte program pro vedení evidence ve skladu. O každém skladovaném materiálu si program pamatuje evidenční číslo, momentální stav zásob a požadovanou minimální zásobu. Uživatel se může informovat o stavu zásob zvoleného materiálu, požadovat seznam materiálu, jehož zásoba poklesla pod požadované minimum, může aktualizovat uložené informace při dodání nebo při vyexpedování materiálu, zaevidovat nový druh materiálu, nebo naopak evidenci některého druhu materiálu zrušit.

340. Sestavte program pro evidenci pacientů v nemocnici, který bude pomáhat sestrám při rozdávání léků. Program si pamatuje u každého pacienta jméno, číslo pokoje, druhy léků, které užívá, dobu, kdy se který lék podává (ráno, v poledne, večer) a v jakém množství. Sestra může zadat dobu a obdrží informaci, do kterých pokojů má komu zanést jaké léky a v jakém množství. Program umožňuje zaevidovat nového pacienta při příchodu a zrušit jeho

evidenci při propuštění z nemocnice. Umožňuje také změnit zvolenému pacientovi rozpis léků.

341. Sestavte program pro evidenci a vyhledávání motorových vozidel. U každého evidovaného vozidla je zaznamenána státní poznávací značka, druh vozidla, barva a majitel. Při zadání některého z těchto údajů (např. jen barva a část SPZ) program zpřístupní seznam vozidel, která odpovídají zadanému popisu.

342. Vytvořte program na evidenci dárců krve. U každého dárce je zaznamenáno jméno, krevní skupina, Rh faktor a datum posledního odběru. Pokud obsluha zadá datum, krevní skupinu a potřebné množství krve, obdrží seznam dárců krve, a případně také náhradníků, kteří budou pozváni k odběru.
Poznámka: Dárce může darovat 200 ml krve jednou za 3 měsíce. Pamatujte, že dárce s některou krevní skupinou může darovat i jiné skupině, uvažujte i Rh faktor.

343. Naprogramujte vedení evidence studentů školy. Vytvořte vhodnou datovou strukturu pro uložení všech potřebných údajů a napište procedury pro vkládání nových studentů, změn osobních údajů, evidenci prospěchu apod. Další procedury budou umožňovat snadné vyhledávání a výpis studentů podle různých kritérií (podle osobních dat, tříd, prospěchu apod.).

344. Naprogramujte vedení evidence vkladních knížek ve spořitelně. Program umožní evidenci vkladních knížek různých typů, zakládání a rušení knížek, vklady, vybírání, výpovědi, připisování úroků a výher, výpis stavu vkladních knížek.

Geometrické úlohy

345. V rovině je dáno N bodů svými kartézskými souřadnicemi, N>3. Nalezněte co nejmenší kružnici takovou, aby všech N bodů leželo uvnitř kružnice (případně na ní). Vypište souřadnice středu a poloměr nalezené kružnice.

346. Dva trojúhelníky v rovině jsou zadány kartézskými souřadnicemi svých vrcholů. Zjistěte, zda některý z nich leží celý uvnitř druhého trojúhelníka.

347. Na kružnici leží N různých bodů A_1, A_2, ..., A_N, $N \geq 4$, v tomto pořadí ve směru hodinových ručiček. Pro každé dva body A_i a A_j na kružnici označíme $|A_iA_j|$ délku oblouku z A_i do A_j měřenou ve směru hodinových ručiček. Jsou dána kladná celá čísla D_1, D_2, ..., D_N udávající délky oblouků mezi dvojicemi sousedních bodů na kružnici: $D_i = |A_iA_{i+1}|$ pro $1 \leq i < N$, $D_N = |A_NA_1|$. Zjistěte, zda mezi danými N body existují čtyři takové, které jsou vrcholy čtverce. Pokud ano, jednu takovou čtveřici bodů vytiskněte (tj. vytiskněte čtveřici indexů těchto bodů). Při výpočtech, které budete v programu provádět, musí být všechny výsledky i průběžné mezivýsledky celá čísla.

348. Mnohoúhelník v rovině je zadán kartézskými souřadnicemi svých vrcholů uspořádaných po obvodu mnohoúhelníka proti směru hodinových ručiček. Zjistěte, zda je daný mnohoúhelník konvexní.

349. Mnohoúhelník v rovině je zadán kartézskými souřadnicemi svých vrcholů uspořádaných po obvodu mnohoúhelníka proti směru hodinových ručiček. Dále jsou dány souřadnice bodu X v rovině. Zjistěte, zda se bod X nachází uvnitř daného mnohoúhelníka. Daný mnohoúhelník nemusí být konvexní.

350. V rovině je dáno N bodů svými kartézskými souřadnicemi, N>3. Nalezněte co nejmenší konvexní mnohoúhelník takový, aby všech N bodů leželo uvnitř tohoto mnohoúhelníka nebo na jeho obvodu (tzn. nalezněte konvexní obal daných bodů). Vypište souřadnice všech vrcholů nalezeného mnohoúhelníka v pořadí, jak po sobě následují na jeho obvodu.

351. Mnohoúhelník v rovině je zadán kartézskými souřadnicemi svých vrcholů uspořádaných po obvodu mnohoúhelníka proti směru hodinových ručiček. Triangulací mnohoúhelníka rozumíme takový soubor jeho úhlopříček, které dělí plochu mnohoúhelníka na samé trojúhelníky. Přitom každá taková úhlopříčka leží celá uvnitř mnohoúhelníka a žádné dvě úhlopříčky se neprotínají. Mnohoúhelník s N vrcholy může mít více různých triangulací, každá z nich je tvořena přesně N-3 jeho úhlopříčkami a dělí plochu mnohoúhelníka na N-2 trojúhelníků. Nalezněte jednu (libovolnou) triangulaci daného mnohoúhelníka a vypište seznam úhlopříček, které ji tvoří. Daný mnohoúhelník nemusí být konvexní.

Grafové algoritmy

352. Číslo N na vstupu představuje počet měst, mezi nimiž zkoumáme možnosti autobusového spojení. Města jsou označena čísly od 1 do N. Mezi některými dvojicemi měst je zavedena přímá autobusová linka. Seznam těchto dvojic měst je zadán. Dále jsou zadána čísla dvou měst, odkud a kam chceme cestovat. Zjistěte, zda existuje nějaké autobusové spojení (přímé nebo s přesedáním) mezi těmito dvěma městy.

353. Číslo N na vstupu představuje počet měst, mezi nimiž zkoumáme možnosti autobusového spojení. Města jsou označena čísly od 1 do N. Mezi některými dvojicemi měst je zavedena přímá autobusová linka. Seznam těchto dvojic měst je uveden na vstupu. Dále je zadáno číslo města, odkud chceme cestovat. Zjistěte, do kterých měst existuje nějaké autobusové spojení (přímé nebo s přesedáním).

354. Na večírku se sešlo N lidí. O některých dvojicích lidí se podařilo zjistit, že se tito lidé narodili ve stejném roce. Na základě zjištěných informací o dvojicích rozdělte všechny lidi na co možná největší skupinky tak, aby bylo jisté, že se všichni lidé z téže skupinky narodili ve stejném roce.

355. Bludiště se skládá z N místností, mezi některými dvojicemi místností vedou chodby. Místnosti jsou očíslovány čísly 1, 2,...N. Bludiště je zadáno v následujícím tvaru: je uveden počet místností N a dále seznam dvojic čísel místností, mezi nimiž je propojení chodbou. Zjistěte, zda v bludišti existuje cesta z místnosti X do místnosti Y a pokud ano, nějakou takovou cestu vypište (jako posloupnost čísel místností, přes které je třeba projít).

356. Je dán systém N letišť označených čísly od 1 do N. Poloha letišť je popsána jejich pravoúhlými (kartézskými) souřadnicemi v rovině. Dále je zadán maximální dolet letadla (tj. vzdálenost, jakou letadlo uletí bez mezipřistání a doplnění pohonných hmot), číslo výchozího a číslo cílového letiště. Nalezněte cestu letadla mezi dvěma zvolenými letišti, která bude nejkratší z hlediska počtu mezipřistání bez ohledu na celkovou délku trasy. Takových

cest s minimálním počtem mezipřistání může být více různých, stačí nalézt jednu libovolnou z nich.

357. Je dán systém N letišť označených čísly od 1 do N. Poloha letišť je popsána jejich pravoúhlými (kartézskými) souřadnicemi v rovině. Dále je zadán maximální dolet letadla (tj. vzdálenost, jakou letadlo uletí bez mezipřistání a doplnění pohonných hmot), číslo výchozího a číslo cílového letiště. Nalezněte cestu letadla mezi dvěma zvolenými letišti, která bude nejkratší z hlediska celkové vzdálenosti, bez ohledu na počet mezipřistání. Takových cest minimální délky může být více různých, stačí nalézt jednu libovolnou z nich.

358. Je dán systém N letišť označených čísly od 1 do N. Poloha letišť je popsána jejich pravoúhlými (kartézskými) souřadnicemi v rovině. Dále je zadán maximální dolet letadla (tj. vzdálenost, jakou letadlo uletí bez mezipřistání a doplnění pohonných hmot), číslo výchozího a číslo cílového letiště. Nalezněte cestu letadla mezi dvěma zvolenými letišti, která bude nejkratší z hlediska celkové vzdálenosti. Pokud takových cest minimálním délky existuje více různých, vyberte z nich cestu s co nejmenším počtem mezipřistání.

359. Je dán systém N letišť označených čísly od 1 do N. Poloha letišť je popsána jejich pravoúhlými (kartézskými) souřadnicemi v rovině. Dále je zadán maximální dolet letadla (tj. vzdálenost, jakou letadlo uletí bez mezipřistání a doplnění pohonných hmot), jeho cestovní rychlost, čas potřebný k mezipřistání a doplnění pohonných hmot, číslo výchozího a číslo cílového letiště. Nalezněte cestu letadla mezi dvěma zvolenými letišti, která bude nejrychlejší, uvažujeme-li také zdržení letadla při nutných mezipřistáních. Pokud existuje více různých cest, které jsou stejně rychlé, stačí nalézt jednu libovolnou z nich.

Textové výstupy

360. Vytiskněte tabulku malé násobilky v přehledném tvaru.

361. Spočítejte a vytiskněte v přehledném tvaru Pascalův trojúhelník. Pascalův trojúhelník je trojúhelníkové schéma tvořené hodnotami binomických koeficientů (kombinačních čísel). V N-té řadě Pascalova trojúhelníka se nacházejí po řadě všechny hodnoty kombinačních čísel (N nad K) pro K od 0 do N. Velikost trojúhelníka, tj. počet jeho řad N, je zadána.

362. Napište program, který vytiskne kalendář pro zadaný rok. Program musí fungovat pro roky 1600 až 2100. Zvolte vhodný a přehledný tvar výsledného tisku, aby bylo jasné jak je rozdělen rok na měsíce, kolik dní je v každém měsíci a na jaký den v týdnu připadá které datum.

363. Vytvořte program, který počítá a tiskne kondiciogram. Vstupem programu bude datum narození a zadání období, na které má být kondiciogram zpracován. Výsledkem práce programu bude vytištění kondiciogramu ve vhodném přehledném tvaru. Kondiciogram je tvořen třemi křivkami sinusového průběhu, které charakterizují kondici člověka po stránce fyzické, emocionální a intelektuální. Všechny tři křivky začínají při narození člověka od nuly směrem do kladných hodnot a mají délku periody 23 dní (fyzický cyklus), 28 dní (emocionální cyklus) a 33 dní (intelektuální cyklus). Je-li křivka kondiciogramu v kladné fázi, má prý člověk dobré a úspěšné dny, je-li v záporné fázi, má dny špatné. Nejhorší jsou tzv. kritické dny, které nastávají vždy v okamžiku, kdy některá z křivek přechází z kladné do záporné nebo ze záporné do kladné fáze (tj. má nulovou hodnotu). V kondiciogramu člověka je tedy každý den charakterizován třemi údaji určujícími, v jaké fázi je každá ze tří křivek kondiciogramu.

364. Vytvořte soustavu procedur sloužících ke kreslení jednoduchých obrázků v rastru tiskárny. Kreslicí systém má umožňovat volbu měřítka na osách, snadné vykreslení bodů, úseček, přímek, kružnic apod. pomocí zadaných souřadnic. S pomocí vytvořené soustavy procedur nakreslete nějaký jednoduchý obrázek.

Práce s grafikou

V této části sbírky jsou obsaženy úlohy různé obtížnosti a různého zaměření, ale s jednou společnou vlastností. Výsledek práce programu má být graficky prezentován na obrazovce osobního počítače nebo grafického terminálu. Úlohy tohoto typu jsou soustředěny do samostatné kapitoly proto, že k jejich vyřešení nestačí standardní prostředky jazyka Pascal, případně jiného běžného programovacího jazyka. V definicích dnes používaných programovacích jazyků se s prostředky pro grafický výstup nepočítalo, neboť jazyky byly původně určeny pro zpracování úloh v dávkovém režimu na velkých počítačích. Možnosti grafického výstupu jsou navíc velmi závislé na vlastnostech použité výpočetní techniky. Překladače programovacích jazyků určené pro osobní počítače zpravidla obohacují prostředky jazyka kromě jiného také o příkazy pro ovládání grafiky na příslušném typu počítače.

Podkapitola věnovaná grafickému výstupu je pouze velmi krátkou orientační ukázkou několika úloh, v nichž je možné a vhodné grafiku na počítači využít. První tři zde uvedené úlohy se opakují z jiné části sbírky (Geometrické úlohy) a jsou pouze doplněny o grafické vykreslení spočitaného výsledku.

365. V rovině je dáno N bodů svými kartézskými souřadnicemi, N>3. Nalezněte co nejmenší kružnici takovou, aby všech N bodů leželo uvnitř kružnice (případně na ní). Vykreslete daných N bodů i nalezenou kružnici.

366. V rovině je dáno N bodů svými kartézskými souřadnicemi, N>3. Nalezněte co nejmenší konvexní mnohoúhelník takový, aby všech N bodů leželo uvnitř tohoto mnohoúhelníka nebo na jeho obvodu (tzn. nalezněte konvexní obal daných bodů). Vykreslete daných N bodů i nalezený mnohoúhelník.

367. Mnohoúhelník v rovině je zadán kartézskými souřadnicemi svých vrcholů uspořádaných po obvodu mnohoúhelníka proti směru hodinových ručiček. Triangulací mnohoúhelníka rozumíme takový soubor jeho úhlopříček, které dělí plochu mnohoúhelníka na samé trojúhelníky. Přitom žádné dvě z těchto úhlopříček se

neprotínají. Mnohoúhelník s N vrcholy může mít více různých triangulací, každá z nich je tvořena přesně N-3 jeho úhlopříčkami a dělí plochu mnohoúhelníka na N-2 trojúhelníků. Nalezněte jednu (libovolnou) triangulaci daného mnohoúhelníka. Mnohoúhelník i nalezenou triangulaci vykreslete. Daný mnohoúhelník nemusí být konvexní.

368. Dva trojúhelníky v rovině jsou zadány kartézskými souřadnicemi svých vrcholů. Zjistěte, zda některý z nich leží celý uvnitř druhého trojúhelníka. Pokud ano, vykreslete oba trojúhelníky s vyplněnou oblastí, která náleží vnějšímu, ale nikoliv vnitřnímu trojúhelníku. Jestliže tomu tak není, vykreslete trojúhelníky a vyplněním odlište oblasti náležející každému z nich, popř. oběma zároveň.

369. Jsou dána tři čísla představující délky hran kvádru. Zobrazte tento kvádr v rovnoběžném promítání pod úhlem 45°.

370. Jsou dány souřadnice vrcholů dvou kvádrů. Oba kvádry mají všechny své stěny rovnoběžné se souřadnicovými osami a jsou disjunktní (nemají žádný společný bod). Zobrazte dvojici kvádrů v rovnoběžném promítání pod úhlem 45°. Nekreslete neviditelné hrany a zakryté části kvádrů.

371. Pro dané číslo N nakreslete šachovnici NxN. Pole šachovnice vybarvěte střídavě dvěma barvami a popište řádky a sloupce šachovnice obvyklým způsobem (čísly a písmeny). Můžete předpokládat vhodné omezení maximální přípustné hodnoty N.

372. Je dána šachová pozice, tj. seznam figur bílého a černého s jejich umístěním na šachovnici. Nakreslete šachovnici s umístěnými šachovými figurami.

373. Je dán počet zobrazovaných údajů a hodnoty, které máme přehledně zobrazit. Vytvořte sloupcový diagram pro zadanou soustavu zobrazovaných hodnot. Volte vhodné měřítko na obou osách (v závislosti na zobrazovaných údajích), aby výsledný diagram dobře využíval plochu.

374. Je dán počet zobrazovaných údajů a hodnoty, které máme přehledně zobrazit. Vytvořte kruhový diagram pro zadanou soustavu zobrazovaných hodnot.

375. Nakreslete kruhové hodiny s ručičkami ukazujícími čas, který je zadán.

376. Jsou dána čtyři reálná čísla X1, X2, Y1, Y2 a reálná funkce jedné proměnné. Nakreslete graf této funkce na intervalu <X1,X2>, přičemž funkční hodnota se zobrazí pouze tehdy, jestliže padne do intervalu <Y1,Y2>. Graf funkce opatřete číselnými osami s měřítkem.

377. Jsou dána dvě reálná čísla X1, X2 a reálná funkce jedné proměnné. Nakreslete graf této funkce na intervalu <X1,X2>. Podle zadané funkce a intervalu <X1,X2> vhodně zvolte umístění počátku soustavy souřadnic a měřítka na obou osách tak, aby graf funkce dobře využíval plochu (tj. aby se na ni vešel, a přitom aby nebyl příliš malý). Nakreslený graf funkce opatřete číselnými osami s měřítkem.

378. Zvolte vhodný způsob zápisu notového záznamu v textovém souboru a napište program, který graficky vykreslí notový zápis podle textu zadaného na vstupu. Dovolí-li vám to technické možnosti vašeho počítače, můžete doplnit grafický výstup ještě odpovídajícím zvukem.

379. Naprogramujte kreslicí systém sloužící k malování jednoduchých čárových kreseb na obrazovce. Systém musí umožňovat snadno kreslit například body, úsečky, kružnice, oblouky, volit barvu podkladu a kreslených čar apod. Zvolte si sami jednoduchý "programovací jazyk" pro zápis kreslicích příkazů. Vytvořený program bude číst ze vstupu posloupnost těchto příkazů a bude je interpretovat, tj. bude na obrazovce kreslit požadovaný obrázek.

Hry

380. Nalezněte takovou cestu šachového koně po šachovnici daných rozměrů, při níž bude každé pole šachovnice navštíveno právě jednou. Velikost šachovnice a souřadnice počátečního pole jsou vstupními daty. Stačí nalézt jeden možný průchod šachovnicí.

381. Napište program, který na šachovnici rozmístí co nejvíce šachových dam tak, aby se vzájemně neohrožovaly. Úlohu řešte pro klasickou šachovnici 8x8 polí nebo pro obecnou obdélníkovou šachovnici NxM polí, přičemž údaje N, M jsou zadány (můžete předpokládat nějaké vhodné omezení jejich povolených hodnot).

382. Napište program, který na šachovnici rozmístí co nejméně šachových dam tak, aby tyto dámy obsazovaly nebo ohrožovaly všechna pole šachovnice. Úlohu řešte pro klasickou šachovnici 8x8 polí nebo pro obecnou obdélníkovou šachovnici NxM polí, přičemž údaje N, M jsou zadány.

383. Na šachovnici o rozměrech NxM polí jsou některá pole obsazena. Šachový král se pohybuje po šachovnici tak, že jedním tahem přejde z pole, na kterém momentálně stojí, na některé z osmi sousedních polí (svisle, vodorovně nebo šikmo). Může samozřejmě vstoupit pouze na volné pole. Napište program, který pro libovolnou dvojici volných polí (výchozí a cílové) najde cestu králem z výchozího pole na pole cílové. Vstupem programu bude seznam souřadnic obsazených polí, souřadnice výchozího a cílového pole.

384. Na šachovnici daných rozměrů NxM se pohybuje šachový kůň. Napište program, který pro zadaná pole (X1,Y1) a (X2,Y2), kde X1, X2 jsou celá čísla od 1 do N a Y1, Y2 celá čísla od 1 do M, určí nejmenší možný počet tahů potřebných na přesunutí šachového koně z výchozího pole (X1,Y1) na cílové pole (X2,Y2). Například pro M=N=8 a pole (2,2) a (2,3) program vypíše číslo 3.

385. Ve čtverečkové síti tvaru obdélníka jsou některé čtverečky vybarveny. Síť představuje bludiště, vybarvené čtverečky tvoří

stěny. Jsou dány rozměry bludiště, umístění stěn a souřadnice výchozího a cílového políčka. Nalezněte cestu bludištěm z výchozího na cílové pole, pokud taková cesta existuje.

386. Ve čtverečkové síti tvaru obdélníka jsou některé čtverečky vybarveny. Síť představuje bludiště, vybarvené čtverečky tvoří stěny. Stěny jsou také po celém obvodu bludiště s výjimkou dvou políček: prostředního políčka levého okraje a prostředního políčka pravého okraje. Jsou dány rozměry bludiště a umístění stěn. Nalezněte cestu bludištěm z levého do pravého otvoru ve vnější stěně bludiště.

387. Napište program, který s vámi bude hrát hru zvanou "Master Mind" nebo také "Logik". Cílem hry je určit uspořádanou čtveřici barev (barvy se v ní mohou opakovat), kterou program předem náhodně zvolil. Hráč postupně předkládá hádané čtveřice barev a program je hodnotí. Hodnocení spočívá v tom, že program sdělí počet správných barev umístěných ve čtveřici na správném místě a počet správných barev, ale na špatném místě ve čtveřici. Počet barev používaných ve hře je předem pevně zvolen (např. 6). Barvy můžete označovat čísly 1,2,3 atd. Například je-li hledanou čtveřicí barev 1 1 3 4 a hráč zvolil 1 4 1 1, odpověď programu bude znít: 1 barva uhádnuta na správném místě, další 2 správné barvy na špatném místě.

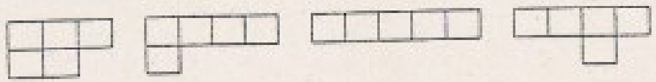
388. Napište program, který bude hrát hru "Master Mind" neboli "Logik". Pravidla hry jsou popsána v předchozí úloze. Úkolem je nyní sestavit program, který bude co nejlepším postupem hádat neznámou čtveřici barev. Hráč si vymyslí tajnou čtveřici, kterou má program uhádnout. Program potom "hádá" a od hráče dostává odpovědi, jak byl úspěšný.

389. V místnosti o rozměrech 2x3 metry je 5 skříněk. Skříňky jsou očíslovány čísly 1 až 5, každá má velikost 1x1 metr a je umístěna v jednom ze čtverců A, B, C, D, E, F. Zbývající šestý čtverec je volný.

A	B	C
D	E	F

Počáteční umístění skříněk ve čtvercích je zadáno. Dále je dáno požadované cílové rozmístění skříněk. Skříňky se mohou přesunovat pouze na sousední políčka (svisle nebo vodorovně). Zjistěte, zda je možné přesouváním skříněk z počátečního rozestavení dosáhnout požadovaného rozmístění, a kolik je na to třeba provést přesunů.

390. Pentamino je hra, při níž se čtverečková síť daných rozměrů pokrývá hracími kameny. Každý hrací kámen je tvořen pěti čtverečky. Existují kameny s různým uspořádáním pětic čtverečků. Kameny se pokládají podle dělicích čar čtverečkové sítě, nikde se nesmějí překrývat a nesmějí přesahovat ven z hrací plochy.
Příklad hracích kamenů:



Je dána velikost obdélníkové hrací plochy a počet každého z typů hracích kamenů. Kameny je možné natáčet, ale ne převracet; jestliže se zrcadlový obraz kamene liší od původního kamene, považuje se za jiný kámen. Zjistěte, zda je možné pomocí daných kamenů hrací plochu zcela pokrýt. Jestliže ano, vytiskněte takové pokrytí. Existuje-li více různých řešení, stačí nalézt jedno libovolné.

391. V předchozí úloze jsou popsána pravidla hry pentamino. Při stejném zadání a vstupních údajích zjistěte, zda je možné všechny dané hrací kameny umístit na hrací plochu. Hrací plocha přitom nemusí být plně pokryta. Je-li to možné, jedno takové rozmístění kamenů vytiskněte.

392. Naprogramujte Conwayovu hru Life. Vyřešte vstup úvodní pozice (počáteční konfigurace), zadání požadovaného počtu generací, výpis každé generace a překročení mezí zvolené hrací plochy.

Hra Life je model života bakterií. Hraje se v rovině ve čtverečkové síti. Každé políčko sítě představuje buňku, která může být živá nebo mrtvá (tj. bakterie v ní žije nebo nežije). Počáteční konfigurací je výchozí rozmístění živých buněk v této síti. Stav buňky v následujícím kroku závisí na počtu živých buněk v jejím okolí (okolím buňky rozumíme 8 sousedních políček) a je dán tzv. zákony života. Živá buňka přežije do následujícího kroku, jestliže v jejím okolí jsou dvě nebo tři živé buňky, jinak zahyne na osamocení (jestliže v jejím okolí nejsou živé buňky nebo je tam jen jedna živá buňka), nebo na přemnožení (jestliže v jejím okolí jsou aspoň čtyři živé buňky). V mrtvé buňce vznikne nová bakterie, jestliže jsou v jejím okolí přesně tři živé buňky. V jednom kroku mění svůj stav všechny buňky najednou - tak vznikne nová generace.

393. V předchozí úloze je popsána hra Life. Některé konfigurace této hry jsou cyklické, tzn. po několika generacích se opakují. Napište program, který zjistí, zda je daná konfigurace cyklická. Můžete předpokládat vhodné omezení počtu generací, během nichž se musí cyklická konfigurace zopakovat (např. nejvýše 50 generací).

394. V předchozích dvou úlohách je popsána Conwayova hra Life. Naprogramujte variantu hry Life pro život více druhů bakterií v jedné síti. Zvolte vhodná pravidla určující, za jakých podmínek bakterie přežívá do další generace a za jakých podmínek v dosud mrtvé buňce vzniká nová bakterie. Pravidla musí určovat i zánik a vznik nových bakterií v místě vzájemného dotyku konkurujících si druhů. Průběh růstu populace po jednotlivých generacích zobrazujte (odlište jednotlivé druhy, například barevně).

395. Jsou dány objemy tří nádob a informace, která z nádob je na začátku plná a která prázdná. Alespoň jedna nádoba je plná a alespoň jedna je prázdná. Dále je dáno množství kapaliny, které máte pomocí těchto nádob odměřit. Určete, zda je toto odměření možné, a pokud ano, určete postup přelévání kapaliny. Nemáte k disposici žádnou další kapalinu než tu, která byla na začátku v plných nádobách. Žádnou kapalinu nesmíte během přelévání mezi nádobami vylít.

396. Jsou dány objemy dvou nádob a požadavek, jaké množství vody máte pomocí těchto nádob odměřit. Určete, zda je toto odměření možné, a pokud ano, určete postup přelévání vody. Máte k disposici vodovod jako neomezený zdroj vody a během přelévání můžete jakékoliv množství vody vylévat do odpadu.

397. Sestavte program, který bude řešit šachové dvojtažky. Na vstupu je zadáno rozmistění šachových figur na šachovnici v obvyklém tvaru (tj. vždy druh figury a souřadnice políčka na šachovnici). V zadané pozici je na tahu bílý. Úkolem programu je nalézt takové pokračování šachové partie, při němž dá bílý druhým tahem mat. Pro jednoduchost neuvažujte rošády a braní mimochodem.

398. Napište program, který bude hrát co nejlepším způsobem piškvorky. Program musí umět vygenerovat počáteční pozici, přijímat tahy od hráče, kontrolovat jejich správnost, zobrazovat je a volit podle zvolené strategie své vlastní tahy.

399. Vytvořte program, který bude hrát co nejlepším způsobem dámu. Program musí umět vygenerovat počáteční pozici, přijímat tahy od hráče, kontrolovat jejich správnost, zobrazovat je a volit podle zvolené strategie své vlastní tahy.

400. Vytvořte program, který bude hrát co nejlepším způsobem žravou dámu. Program musí umět vygenerovat počáteční pozici, přijímat tahy od hráče, kontrolovat jejich správnost, zobrazovat je a volit podle zvolené strategie své vlastní tahy.

Návody k řešení úloh

4. Využijte celočíselné dělení šedesáti, resp. 3600.

25. Dejte pozor na dělení nulou. Je-li některé ze zadaných čísel nulové, vypište zprávu, že výraz nelze pro zadaná x, y vyhodnotit.

28. Obě hledané hodnoty zjišťujte zároveň. Porovnejte mezi sebou zvlášť první dvě a zvlášť druhá dvě zadaná čísla. Pak porovnejte větší hodnoty z každé dvojice (dostanete celkově největší číslo) a menší hodnoty z každé dvojice (dostanete nejmenší číslo ze všech).

29. Délky stran trojúhelníka splňují tzv. trojúhelníkové nerovnosti: součet libovolných dvou z nich je větší než délka zbývající strany.

31. Nezapomeňte na případ, kdy A=0. Pak má rovnice buď nekonečně mnoho kořenů (pokud také B=0), nebo nemá kořeny žádné (je-li koeficient B různý od nuly).

32. Spočítejte diskriminant rovnice a před výpočtem kořenů otestujte, zda není záporný. V takovém případě by rovnice neměla žádný reálný kořen. Nezapomeňte také zvlášť vyřešit případ, kdy A=0.

33. Má-li měsíc 28 dní, jsou v něm vždy čtyři pátky. Má-li měsíc 29 dní, je v něm pět pátků jedině tehdy, pokud začíná v pátek, jinak jsou v něm čtyři pátky. Má-li měsíc 30 dní, je v něm pět pátků tehdy, pokud začíná ve čtvrtek nebo v pátek, jinak jsou v něm čtyři pátky. Podobnou úvahu proveďte pro měsíc s 31 dny.

34. Pomocí návodu k předchozí úloze zjistěte počet sobot a nedělí v měsíci.

44. Aritmetický průměr je možné počítat pouze v případě, že na vstupu bylo aspoň jedno číslo z intervalu (0,100). Jinak by došlo k dělení nulou a výpočet programu by skončil běhovou chybou.

47. Uvnitř cyklu čtěte a zpracujte vždy trojici čísel.

52. Nestačí vynásobit roční úrokovou míru dobou uložení úspor a o tolik procent zvýšit hodnotu počátečního vkladu! Musíte počítat se složeným úrokováním, tzn. ve druhém roce se již počítají úroky nejen z původního vkladu, ale i z úroků připsaných po prvním roce, atd.

53. Stejně jako v předcházející úloze musíte počítat se složeným úrokováním, tzn. ve druhém a každém dalším roce připisovat na vkladní knížku i úroky z úroků.

56. Je-li číslo n záporné, použijte vztah $x^n = 1/x^{-n}$.

63. Není dobré spočítat zvlášť čitatele a zvlášť jmenovatele zlomku v definici kombinačního čísla. Mohlo by se stát, že i pro dosti malá n, k překročí hodnota čitatele povolený rozsah hodnot typu integer, i když výsledná hodnota kombinačního čísla není příliš velká. Nejlepší je střídavě násobit a dělit v pořadí:

$$n / 1 \cdot (n-1) / 2 \cdot (n-3) / 3 \cdot \cdots \cdot (n-k+1) / k$$

Všimněte si, že při tomto postupu výpočtu jsou všechny mezivýsledky celočíselné.

65. Počet slov věty je o jednu větší než počet mezer.

70. Postupně čtěte čísla, v jedné proměnné si udržujte hodnotu dosud největšího nalezeného čísla a ve druhé počet jejích výskytů v posloupnosti.

72. Postupně čtěte čísla, v jedné proměnné si udržujte hodnotu dosud největšího nalezeného čísla menšího než 100 a ve druhé hodnotu dosud nejmenšího nalezeného čísla většího než 100. Po přečtení všech 30 čísel porovnejte, která z těchto proměnných má hodnotu bližší číslu 100 (popř. zjistite, že jejich vzdálenosti od 100 jsou stejné). Nezapomeňte vhodně ošetřit případ, že mezi zadanými čísly je i číslo 100.

73. Postupně čtěte čísla, v jedné proměnné si udržujte hodnotu dosud největšího nalezeného čísla a ve druhé hodnotu druhého největšího čísla.

74. Postupně čtěte čísla, v jedné proměnné si udržujte hodnotu dosud největšího nalezeného čísla a ve druhé hodnotu zatím nejmenšího nalezeného čísla.

75. Postupně čtěte čísla, v jedné proměnné si udržujte hodnotu dosud nejmenšího nalezeného čísla, ve druhé počet jeho výskytů, ve třetí proměnné hodnotu druhého nejmenšího čísla a v další počet jeho výskytů. Důležité je nezapomenout počítat si průběžně počet výskytů dosud nejmenšího nalezeného čísla! Toto číslo se může časem stát druhým nejmenším a v tom okamžiku bylo již přečteno několik jeho výskytů.

76. Nejvíce se od aritmetického průměru liší nejmenší nebo největší z naměřených hodnot. Stačí jednou projít všechny naměřené hodnoty, přitom spočítat jejich aritmetický průměr a zároveň nalézt minimum a maximum z výsledků měření. Zároveň si musíte zaznamenávat, při kolikátém měření byla naměřena nejmenší a při kolikátém největší hodnota. Pak již stačí porovnat, zda se od aritmetického průměru více liší nalezené minimum nebo maximum.

77. Nezapomeňte, že číslo X může patřit také na začátek nebo na konec dané posloupnosti.

79. Porovnejte první dvě čísla a podle výsledku jejich porovnání rozlište další průběh výpočtu. Je-li například první číslo menší než druhé, může být posloupnost jedině rostoucí, případně neklesající, nebo není monotónní. Postupně čtěte další čísla ze vstupu a každé vždy porovnejte s číslem předchozím. Dokud přicházejí stále větší čísla, posloupnost může být rostoucí. Jakmile jednou přečtete stejné číslo jako bylo předchozí, posloupnost již může být jedině neklesající (zaznamenejte si v pomocné proměnné). Pokud přečtete číslo menší než předcházející, posloupnost není monotónní a výpočet ukončete. Podobně postupujte i v ostatních případech.

80. Je zbytečné zkoušet dělit dané číslo N všemi čísly od 2 do N-1. Stačí otestovat jeho dělitelnost dvěma, a pak všemi lichými čísly od 3 do druhé odmocniny z N.

81. Dané číslo v cyklu celočíselně dělte deseti tak dlouho, dokud je podíl nenulový. Přitom počítejte počet provedených dělení.

82. Dané číslo v cyklu celočíselně dělte deseti tak dlouho, dokud je podíl nenulový. Přitom sčítejte zbytky po prováděném celočíselném dělení deseti.

84. Předem neznáte počet cifer zadaného čísla, takže při postupném celočíselném dělení čísla deseti ještě nevíte, která cifra leží na liché pozici počítáno zleva. Počítejte proto souběžně součet cifer na lichých pozicích zprava a na sudých pozicích zprava. Ten ze součtů, do kterého přispějete naposledy, pak bude hledaným výsledkem.

86. Dané číslo postupně (K-1)krát celočíselně dělte deseti. Tím z něho odstraníte posledních K-1 cifer. Pokud je výsledek po provedených K-1 děleních nulový, dané číslo N mělo méně než K cifer a úloha tedy nemá řešení. V opačném případě je teď hledaná cifra na konci čísla. Získáte ji snadno jako zbytek po celočíselném dělení deseti (operací mod 10).

87. Nejprve určete počet cifer čísla N postupným celočíselným dělením deseti. Hodnotu čísla N si ale předem uložte do pomocné proměnné. Zjistěte, zda má úloha řešení, tj. zda počet cifer čísla N je alespoň K. Pokud ano, snadno spočítáte, kolikátou cifru zprava v čísle N máte nalézt (od počtu cifer čísla N odečtete K-1). Tím je zadaná úloha převedena na lehčí (viz návod k předchozí úloze).

89. Od daného čísla postupně odebírejte zprava jednotlivé cifry (pomocí operací mod a div), dokud zůstává nenulové. Ze získaných cifer přitom zároveň vytvářejte výsledné číslo (postupným násobením deseti a přičítáním cifer).

91. Uvědomte si, co vlastně dvojkový zápis čísla znamená. Má-li číslo K dvojkový zápis $a_n a_{n-1} \dots a_1 a_0$, kde a_i ($i=0,1,\dots,n$) jsou binární cifry 0 nebo 1, platí rovnost:

$$K = a_n \cdot 2^n + a_{n-1} \cdot 2^{n-1} + \dots + a_1 \cdot 2^1 + a_0 \cdot 2^0$$

Pro výpočet hodnoty K z dvojkového zápisu čísla je výhodné upravit si uvedenou rovnost pomocí tzv. Hornerova schématu:

$$K = (\dots ((a_n \cdot 2 + a_{n-1}) \cdot 2 + a_{n-2}) \cdot 2 + \dots + a_1) \cdot 2 + a_0.$$

Stačí pak provést v cyklu N-1 násobení dvěma a stejný počet přičtení nuly nebo jedničky.

92. Nejjednodušší je postupně dělit dané číslo celočíselně dvěma. Jako zbytky při tomto celočíselném dělení dostanete přímo binární cifry tvořící zápis čísla ve dvojkové soustavě. Cifry ovšem dostanete v pořadí od nejnižších řádů, tj. odzadu. Pokud se spokojíte s takovýmto tvarem výsledku, jste hotovi. Jinak je třeba buď použít pole, s jehož pomocí obrátíte pořadí cifer při tisku výsledku, nebo musíte zvolit jiný postup výpočtu. Dvojkový zápis se správným pořadím cifer lze získat i bez použití pole, ale za cenu pomalejšího a složitějšího výpočtu. Nejprve je třeba spočítat počet cifer výsledku (obdobným postupem, jako v předchozím postupu řešení) a dané číslo pak postupně celočíselně dělit vhodnými mocninami dvojky.

93. Nezapomeňte ošetřit případ, že nejdelší souvislý úsek tvořený kladnými čísly je na samém konci vstupní posloupnosti. Dejte také pozor na případ, že daná posloupnost neobsahuje žádné kladné číslo a úloha tedy nemá řešení.

94. Každé přečtené číslo vždy porovnejte s předchozím. Je-li větší, zvyšte o jedničku údaj o délce právě zkoumaného rostoucího úseku. V opačném případě rostoucí úsek skončil a jeho délku porovnejte s průběžně ukládaným dosud nalezeným maximem. Zároveň již máte přečteno první číslo dalšího rostoucího úseku, takže počitadlo délky zkoumaného úseku nastavte na 1 (nikoliv na nulu!). Nezapomeňte ošetřit případ, že nejdelší souvislý rostoucí úsek je na samém konci vstupní posloupnosti.

95. Každé přečtené číslo vždy porovnejte s předchozím. Je-li stejné, zvyšte o jedničku údaj o délce právě zkoumaného úseku stejných čísel. V opačném případě jeden úsek stejných čísel skončil a jeho délku porovnejte s průběžně ukládaným dosud nalezeným maximem. Zároveň již máte přečteno první číslo dalšího úseku, takže počitadlo délky zkoumaného úseku nastavte na 1 (nikoliv na nulu!). Nezapomeňte ošetřit případ, že nejdelší souvislý úsek stejných čísel je na samém konci vstupní posloupnosti.

96. Při jednom průchodu danou posloupností čísel si v jedné proměnné udržujte délku dosud nejdelšího nalezeného úseku stejných čísel a ve druhé počet úseků této délky.

98. Postupně čtěte daná čísla. Vždy od začátku každého nového nenulového úseku musíte začít počítat nejen délku úseku, ale i hodnotu největšího čísla v tomto úseku. Předem totiž nevíte, zda to nebude úsek nejdelší.

103. Postupným dělením deseti získáte jednotlivé cifry daného čísla. K evidenci použitých cifer dobře poslouží například pole indexované od 0 do 9 se složkami typu boolean.

104. Výskyty jednotlivých písmen si počítejte v poli celých čísel indexovaném znaky 'A' až 'Z'.

105. Odpověď může být kladná pouze pro řetězce stejné délky. V takovém případě postupně probírejte jednotlivé znaky prvního řetězce. Každý znak se pokuste vyhledat ve druhém řetězci a pokud tam je, vypusťte ho. Jestliže nějaký znak prvního řetězce ve druhém řetězci nenajdete, je odpověď záporná a výpočet ihned ukončete. Projdete-li úspěšně všechny znaky prvního řetězce, bude odpověď kladná.

Jiný způsob řešení vede k rychlejšímu výsledku, ale vyžaduje použití pomocního pole. V tomto poli celých čísel, které bude indexováno všemi přípustnými znaky, si počítejte výskyty jednotlivých znaků v daných řetězcích. Nejprve všem složkám pole přiřaďte nulu. Potom projděte po znacích prvního řetězce a jim odpovídající prvky pole zvyšujte o jedničku. Dále projděte znaky druhého řetězce a jim odpovídající prvky pole zmenšujte o jedničku. Nakonec zkontrolujte, zda jsou všechny složky pomocného pole opět nulové. Pokud ano, je jeden řetězec permutací druhého, v opačném případě bude odpověď záporná.

106. Všechna čísla si musíte průběžně při čtení ukládat do pole, abyste mohli nejprve zjistit zkoumanou hodnotu posledního z nich, a pak všechna zadaná čísla znovu projít a spočítat hodnoty rovné poslednímu číslu.

107. Čísla dělitelná sedmi si musíte uložit do pole, abyste mohli nejprve vytisknout jejich počet (ten budete znát až po přečtení všech čísel ze vstupu), a pak teprve tato čísla. Ostatní čtená čísla naopak ukládat nemusíte.

108. Dvojnásobky kladných čísel vypisujte hned v průběhu čtení, nekladná čísla si musíte uložit do pole.

110. V pořadí lichá čtená čísla ihned vypisujte, sudá si ukládejte do pole.

111. Při čtení čísel si počítejte průběžnou pozici a do pomocného pole si ukládejte pozice výskytů dosud nejmenšího nalezeného čísla. Vždy při nalezení nového minima obsah tohoto pole smažte.

112. Čísla čtěte ze vstupu a ukládejte je do pole - nezáporná čísla odpředu, záporná odzadu. Obsah celého pole potom vypište od začátku do konce.

Jinou možností řešení je nezáporná čísla vůbec neukládat, ale rovnou je vypisovat.

113. K vyřešení úlohy bude stačit jedno pomocné pole pro uložení 100 celých čísel. Čísla čtěte ze vstupu a ukládejte je do pole - nezáporná čísla odpředu, záporná odzadu. Nezáporná čísla jsou v poli uložena v původním pořadí, ale záporná v opačném pořadí. Vypište proto nejprve počáteční úsek pole s nezápornými čísly odpředu a pak koncový úsek pole se zápornými čísly odzadu.

Druhou možností řešení je nezáporná čísla vůbec neukládat, ale rovnou je vypisovat.

115. Zadané hodnoty není třeba ukládat do pole. Stačí použít tři jednoduché proměnné, v nichž budou v každém okamžiku výpočtu uložena tři naposledy přečtená čísla. Z nich spočítejte aritmetický průměr, vytiskněte ho a přečtěte další číslo ze vstupu.

116. Nejjednodušší řešení využívá pole velikosti N, do kterého se nejprve uloží všech N naměřených hodnot přečtených ze vstupu. Stačí pak v cyklu procházet všechny K-prvkové úseky a počítat jejich průměry. Je ovšem zbytečné počítat vždy znovu celý součet dalšího K-prvkového úseku. Lepší je zaznamenat si součet předchozího úseku a nový součet spočítat z něho (odečtením jeho prvního prvku a přičtením jednoho nového čísla). Zdaleka ale

není třeba použít pole velikosti N. Stačí pracovat s menším polem velikosti K, ve kterém je v každém okamžiku výpočtu uloženo K naposledy přečtených čísel (na začátku práce se do něho uloží prvních K čísel). Z nich se vždy spočítá aritmetický průměr, vytiskne se a přečte se další číslo ze vstupu. Nyní můžete všechna čísla v poli posunout o 1 místo doleva, tím se ztratí "nejstarší" z uložených čísel a uvolní se volné místo na konci pole pro uložení právě přečteného čísla. Jiný, ještě lepší postup nepotřebuje ani opakované posouvání všech čísel uložených v poli. Právě přečtené číslo je totiž možné uložit do pole přímo na místo dosud "nejstaršího" uloženého čísla. K tomu je ale nutné zaznamenávat si informaci, které uložené číslo v poli je "nejstarší".

117. Jsou možné různé postupy řešení. Nejjednodušší je průběžně si ukládat všechna přečtená čísla do pole a vždy po přečtení dalšího čísla zkontrolovat, zda již takové číslo není uloženo v poli. Výpočet zrychlíte, jestliže přečtená čísla nejprve uložíte do pole, a pak je v něm setřídíte podle velikosti. Případná opakující se stejná čísla potom snadno vyhledáte při jednom postupném průchodu polem (jsou umístěna vedle sebe).

118. Čtěte čísla ze vstupu a průběžně si udržujte seznam různých hodnot. Po přečtení dalšího čísla zkontrolujte, zda se již v posloupnosti vyskytlo. Pokud ne, přidejte ho do seznamu. Jinou možností řešení je uložit všechna čísla do pole a setřídit je podle velikosti. Výsledek pak získáte snadno při jednom průchodu setříděným polem (opakující se výskyty čísel jsou vedle sebe).

119. Čtěte čísla ze vstupu, průběžně si udržujte seznam těch z nich, která se vyskytla právě jednou, a vedle toho zvlášť seznam těch, která se již vyskytla alespoň dvakrát. Po přečtení dalšího čísla zkontrolujte, kolikrát se již v posloupnosti vyskytlo. Pokud ještě vůbec ne, přidejte ho do seznamu čísel s jedním výskytem, pokud zatím jen jednou, přeřaďte ho do druhého seznamu. Jinou možností řešení je uložit všechna čísla do pole a setřídit je podle velikosti. Výsledek pak získáte při jednom průchodu setříděným polem (opakující se výskyty čísla jsou vedle sebe).

120. Postupujte stejně jako v předchozí úloze.

121. Využijte rady z návodu k předcházejícím úlohám.

122. Čtěte čísla ze vstupu, průběžně si udržujte seznam těch z nich, která se vyskytla zatím jen jednou (navíc ne nutně všech, stačí jen těch z nich, která jsou tak malá, že ještě mohou ovlivnit výsledek), a vedle toho si zvlášť pamatujte hodnotu M nejmenšího čísla, které se již vyskytlo alespoň dvakrát. Jestliže přečtete číslo větší než momentální hodnota M, nic s ním nemusíte dělat, neboť takové číslo nemůže nijak ovlivnit výsledek. Jinak po přečtení dalšího čísla ze vstupu nejprve zkontrolujte, zda je obsaženo ve vytvářeném seznamu. Pokud ne, přidejte ho do seznamu čísel s jedním výskytem. Jestliže se právě přečtené číslo již v posloupnosti vyskytlo dříve, tj. našli jste ho v seznamu, můžete ho ze seznamu odstranit a patřičně zmenšit hodnotu M. Jinou možností řešení je uložit všechna čísla do pole a setřídit je podle velikosti. Výsledek pak získáte při jednom průchodu setříděným polem (opakující se výskyty čísla jsou vedle sebe).

123. Využijte návod k předchozí úloze.

124. Nezapomeňte, že symetrický úsek může být liché nebo sudé délky, tzn. jeho střed leží buď v místě některého prvku posloupnosti, nebo mezi dvěma sousedními prvky. Symetrii každého úseku je výhodné prověřovat vždy od jeho středu k okrajům. Možnými středy jsou všechny prvky dané posloupnosti a všechny pozice mezi dvěma sousedními prvky.

126. Čísla tvořící posloupnost čtěte ze vstupu, ukládejte si je do pole, zároveň hned počítejte délky souvislých rostoucích úseků a evidujte pozice v dané posloupnosti čísel, na nichž začínají jednotlivé rostoucí úseky dosud největší nalezené délky. Po přečtení a uložení celé posloupnosti čísel pak již stačí projít vytvořenou evidenci a podle odkazů do posloupnosti vypsat rostoucí úseky maximální délky.

128. Pro větší přehlednost a srozumitelnost programu je výhodné použít pole, ve kterém jsou uloženy počty dní v jednotlivých měsících. Pokud ovšem nepředpokládáte opakování výpočtu pro různá data, vede toto řešení k pomalejšímu výpočtu, než

kdybyste příslušné číselné konstanty uvedli přímo v příkazech programu.

129. Použijte pole, ve kterém jsou uloženy počty dní v jednotlivých měsících. Nezapomeňte určit, zda se jedná o rok přestupný.

130. Použijte pole, ve kterém jsou uloženy počty dní v jednotlivých měsících.

131. V uvedeném rozmezí jsou přestupné všechny roky, jejichž letopočet je dělitelný čtyřmi, s výjimkou roků 1800, 1900 a 2100.

132. Určete počet dní mezi zkoumaným datem a nějakým pevným datem, o němž víte, na který den v týdnu připadne (např. dnešním dnem). Dejte pozor na přestupné roky.

134. Nejmenší vzdáleností bodu X od daného obdélníka je buď kolmá vzdálenost bodu X od některé strany obdélníka, nebo vzdálenost bodu X od některého vrcholu obdélníka.

136. Použít Eratosthenovo síto je lepší než testovat každé číslo od 2 do N, zda je prvočíslem.

145. Postupné mechanické ověřování každé dvojice čísel A, B menších než dané N je velmi pomalé, neboť pro mnohá čísla tak musíme opakovaně zjišťovat součet jejich dělitelů. Výhodnější je spočítat si předem pro každé číslo od 1 do N součet kladných dělitelů (vždy s výjimkou jeho samotného), tento součet si uložit do pomocného pole a dvojice spřátelených čísel pak vyhledávat s využitím spočítaných hodnot uložených v poli.

146. Můžete spočítat všechna prvočísla od 2 do B pomocí Eratosthenova síta, a pak z nich vybrat ta, která jsou větší nebo rovna A a přitom končí číslicí 1. Jinou možností je testovat jednotlivé všechna taková čísla z intervalu <A,B>, která končí číslicí 1, zda jsou prvočísly.

155. Je zbytečné procházet všechna čtyřciferná čísla a každé testovat, zda obsahuje aspoň tři stejné cifry. Rychlejší je vytvářet přímo všechna čísla požadovaného tvaru.

156. Je zbytečné procházet všechna čísla menší než N a každé testovat, zda je tvořeno pouze ciframi 2, 3 a 5. Rychlejší je vytvářet přímo všechna čísla požadovaného tvaru.

157. Nepoužívejte tři do sebe vnořené cykly pro vyzkoušení všech možných hodnot čísel a,b,c. Ke zkoumaným hodnotám čísel a,b stačí příslušné c dopočítat.

158. Abyste zbytečně nevytvářeli rozklady lišící se pouze pořadím sčítanců, zvolte si uspořádání hodnot sčítanců v rozkladu. Hledejte takové trojice čísel x, y, z, pro které platí x+y+z=N a zároveň x≤y≤z. Stačí uvažovat možné hodnoty x od 1 do N/3. Pro každé takto zvolené x připadají v úvahu hodnoty y pouze z rozmezí od x do (N-x)/2. Zbývající člen rozkladu se již dopočítá ze vztahu z=N-x-y.

159. Hodnoty u nás platných platidel jsou zvoleny tak, že pro zaplacení libovolné částky nejmenším možným počtem platidel je výhodné použít vždy platidlo co nejvyšší hodnoty (a stejně pak postupovat se zaplacením zbývající částky). Díky tomu je zadaná úloha velmi snadná. Uvedené pravidlo ale obecně neplatí pro libovolnou sadu platidel. Kdybyste například měli mince o hodnotách 1 Kčs, 3 Kčs a 4 Kčs, k zaplacení částky 6 Kčs není nejvýhodnější použít minci 4 Kčs a pak dvakrát 1 Kčs, lepší je použít dvě mince po 3 Kčs.

160. V čísle N určete první cifru zprava, která je menší než některá z cifer po ní následujících. Počáteční úsek zápisu čísla M až do této cifry zůstane stejný jako v N, tato cifra se nahradí nejmenší vyšší cifrou, která leží v zápisu čísla N vpravo od ní, zbytek zápisu čísla M získáte tak, že všechny zbývající cifry, které ještě máte použít, uspořádáte vzestupně. Jsou-li cifry daného čísla uspořádány sestupně, úloha nemá řešení.

161. Všimněte si, že pro některá čísla úloha nemá řešení - například pro K=5 a dané číslo 98765.

162. Je-li dané číslo tvořeno K devítkami, úloha nemá řešení.

179. Zamyslete se nad tím, zda sedlový bod matice musí vždy existovat, a uvažte, kolik různých sedlových bodů může mít daná matice.

183. Mocninu můžete počítat násobením v cyklu nebo použít standardní funkce exp a ln.

185. Dekadické číslice mají kódy uspořádané vzestupně od '0' do '9' a tyto kódy tvoří v tabulce znakových kódů souvislou řadu.

186. Jestliže znáte tabulku kódů, se kterou pracuje váš počítač, řešení úlohy se může značně zjednodušit. V případě kódu ASCII, který je používán také na osobních počítačích, využijte skutečnost, že malá i velká písmena anglické abecedy tvoří v tabulce kódů souvislé úseky uspořádané od 'a' do 'z', resp. od 'A' do 'Z'.

190. Funkce počítající největšího společného dělitele dvou čísel využívá Eukleidův algoritmus. V proceduře na krácení zlomků pak stačí spočítat největšího společného dělitele čitatele a jmenovatele a touto hodnotou čitatele i jmenovatele vydělit. Procedura bude mít dva parametry předávané odkazem, v nichž obdrží hodnoty čitatele a jmenovatele kráceného zlomku a vrátí nové hodnoty čitatele a jmenovatele zlomku po zkrácení.

191. Úlohu je možné řešit pomocí rozkladu obou čísel na prvočinitele. Lepší je ale využít skutečnost, že je snadné nalézt největšího společného dělitele daných dvou čísel pomocí Eukleidova algoritmu, a dále že nejmenší společný násobek dvou čísel je roven součinu obou čísel vydělenému jejich největším společným dělitelem.

194. Procedura má dva vstupní parametry předávané hodnotou a dva výstupní parametry, které musí být předávány odkazem.

196. Využijte funkci z předchozí úlohy na zjišťování, zda je rok přestupný.

197. Oba parametry jsou zároveň vstupní i výstupní. Musí být proto předávány odkazem.

204. Nemůžete kopírovat celý soubor znak po znaku mechanicky v jednom cyklu, ve výsledném souboru by pak nebylo zachováno rozdělení textu na řádky.

205. Do vytvářeného výsledného souboru zkopírujte nejprve obsah prvního souboru a hned za něj obsah druhého souboru.

211. Soubor čtěte po řádcích, v každém řádku spočítejte slova.

213. Soubor čtěte po slovech, průběžně si ukládejte dosud nejdelší nalezené slovo a jeho délku.

217. Pro ukládání počtů jednotlivých písmen použijte pole celých čísel indexované písmeny.

220. Počet čísel ve vstupním souboru nemusí být dělitelný třemi.

221. Opakující se výskyty téhož čísla následují v uspořádaném souboru bezprostředně po sobě, takže jejich vyhledání je snadné.

223. Použijte postup označovaný jako "slévání". Z každého souboru přečtěte první číslo, porovnejte je a menší z nich zapište do výstupního souboru. Potom se posuňte na další číslo v tom souboru, ve kterém bylo menší z obou porovnávaných čísel. Celý postup opakujte tak dlouho, dokud nedojdete na konec jednoho ze vstupních souborů. Na závěr zbývá překopírovat do výstupního souboru ještě všechna čísla z toho vstupního souboru, který ještě není dočten do konce.

227. Při zpracování textu je třeba dát velký pozor na ošetření případu, že se v souboru nachází řetězec tvaru např. 'PEPEPES'.

230. Seznam můžete vytvářet buď od začátku s připojováním nových prvků na konec seznamu, nebo odzadu a nové prvky připojovat vždy na začátek seznamu. Druhá z uvedených možností je o něco jednodušší.

231. Čísla čtěte ze vstupu a hned je vkládejte do nově vytvářených prvků seznamu. Seznam vytvářejte odzadu, tj. nové prvky připojujte vždy na začátek seznamu.

232. Čísla čtěte ze vstupu a hned je vkládejte do nově vytvářených prvků seznamu. Seznam vytvářejte odpředu, tj. nové prvky připojujte vždy na konec seznamu. K tomu je výhodné udržovat si stále pomocný ukazatel na dosud poslední prvek vytvářeného seznamu. Vložení prvního čísla do seznamu se ale musí provést odlišným způsobem.

234. Vynechání prvku ze začátku spojového seznamu se provádí jinak než vynechání jiného prvku. Proto je vhodné nejprve ve zvláštním cyklu vypustit všechny prvky s číslem dělitelným pěti, které jsou umístěny na začátku seznamu.

237. Nejjednodušší postup řešení vyžaduje projít daným spojovým seznamem dvakrát. Při prvním průchodu se vyhledá vypouštěný prvek, při druhém průchodu se najde jeho předchůdce a patřičný prvek se ze seznamu vypustí. Lepší je řešení, které vystačí s jedním průchodem. Při postupném průchodu spojovým seznamem si v takovém případě stále musíte udržovat odkaz na předchůdce prvního takového prvku, který obsahuje dosud největší nalezené číslo.

238. Viz návod k předchozí úloze. Jediná změna spočívá v tom, že při postupném průchodu spojovým seznamem si stále musíte udržovat odkaz na předchůdce posledního takového prvku, který obsahuje dosud největší nalezené číslo.

239. Máte v podstatě dvě možnosti, jak řešit úlohu. Při průchodu daným spojovým seznamem si můžete vytvořit pomocný seznam odkazů na předchůdce všech takových prvků, které obsahují největší nalezené číslo. Pomocí tohoto pomocného seznamu potom můžete příslušné prvky vypustit. Přitom je třeba zvlášť ošetřit případ, že se největší z čísel nachází také v prvním prvku seznamu (tento prvek nemá předchůdce). Druhou možností je projít daný spojový seznam dvakrát. Při prvním průchodu pouze určíte největší číslo v seznamu, při druhém průchodu vypustíte všechny prvky seznamu, které ho obsahují. Druhý postup řešení může vést k o něco pomalejšímu výpočtu, ale lépe se programuje.

242. Vždy po vložení nového prvku do seznamu musíte dát pozor, abyste se pro další práci přesunuli až za něj, tzn. na další prvek původního seznamu.

243. Vždy po vložení nového prvku do seznamu musíte dát pozor, abyste se pro další práci přesunuli až za něj, tzn. na další prvek původního seznamu.

245. První seznam s lichými čísly bude vytvářen odpředu, druhý seznam se sudými čísly odzadu.

246. První seznam s písmeny bude vytvářen odpředu, druhý seznam s číslicemi odzadu.

249. Procedura bude mít tři parametry: ukazatel na začátek výchozího spojového seznamu, celočíselný parametr K a pole ukazatelů na začátky vytvářených seznamů.

254. Ze začátku daného seznamu postupně odpojujte jednotlivé prvky a vytvářejte z nich nový seznam. Přesouvané prvky zapojujte vždy na začátek nově vytvářeného seznamu. Po rozebrání celého daného seznamu bude vytvořený výsledný seznam obsahovat přesně stejné prvky, ale v opačném pořadí, neboť byl vytvářen odzadu.

255. Na rozdíl od prostého spojení dvou seznamů zde musíte zabránit opakovanému výskytu téže hodnoty ve výsledném seznamu.

257. Na rozdíl od předchozích úloh stačí jeden souběžný průchod oběma danými seznamy.

258. Pro potřeby sčítání je výhodné zvolit si takovou orientaci spojového seznamu, aby na jeho začátku byly cifry nejnižších řádů a na konci cifry nejvyšších řádů. Během sčítání pak nezapomeňte na přenosy do vyššího řádu. Pamatujte na to, že každé z čísel může mít jiný počet cifer.

259. Je možné použít spojové seznamy jak s orientací od nejnižších řádů čísla k nejvyšším (tj. s ciframi uloženými odzadu), tak i s orientací opačnou (tj. cifry uložené odpředu, od nejvyššího

řádu k nejnižšímu). V obou případech stačí projít daná čísla pouze jednou.

260. Není-li znám žádný odhad omezující počet cifer sčítaných čísel, je třeba reprezentovat je pomocí spojových seznamů. V prvcích seznamů mohou být uloženy jednotlivé cifry nebo případně pro úsporu paměti skupiny cifer. Pro potřeby sčítání je výhodné zvolit si takovou orientaci spojového seznamu, aby na jeho začátku byly cifry nejnižších řádů a na konci cifry nejvyšších řádů. Během sčítání pak nezapomeňte na přenosy do vyššího řádu.

261. Použijte návod k předchozí úloze.

264. Nejlepší je zařazovat nové prvky se čtenými čísly do vytvářeného seznamu tak, aby byl seznam hned od začátku stále setříděný podle velikosti.

265. Použijte některý z běžných třídicích algoritmů. Nejlépe se zde hodí třídění přímým výběrem nebo přímé zatřiďování.

268. Úlohu je možné řešit jedním průchodem daným seznamem. Zrcadlový obraz řetězce si nejprve vytvořte do druhého seznamu, a ten pak vcelku připojte na konec původního seznamu.

273. Násobte běžným způsobem každý člen jednoho činitele každým členem druhého činitele. Před vytvořením nového prvku pro uložení výsledku jednoho takového elementárního násobení musíte zkontrolovat, zda ve vytvářeném seznamu, který bude reprezentovat součin polynomů, není již obsažen prvek se stejným exponentem. V takovém případě nezařadíte do seznamu nový prvek, ale jen změníte hodnotu koeficientu v prvku s příslušným exponentem. (Navíc se po takové změně koeficientu může dokonce stát, že nová hodnota koeficientu bude nulová a celý člen budete muset z polynomu odstranit.)

274. Při průchodu stromem si vytvářejte pomocnou frontu odkazů na jednotlivé uzly stromu (v pořadí "po vrstvách").

276. V pomocných proměnných si průběžně udržujte dvě dosud největší přečtená čísla. Do vytvářeného stromu vkládejte pouze ta čísla, která tam jistě patří a nebude třeba je později vypouštět.

279. Výpis hodnot uzlů v požadovaném pořadí podle velikosti získáte při běžném "středním" průchodu stromem (tzv. průchod inorder).

280. Použití rekurze nahradíte tím, že si naprogramujete zásobník.

281. Použijte rekurzivní proceduru. Vytvořte kořen stromu, uložte do něho hodnotu ((N+1) div 2), a dále vytvořte levý a pravý podstrom s hodnotami uzlů 1 až ((N+1) div 2)-1 v levém podstromu a s hodnotami ((N+1) div 2)+1 až N v pravém podstromu.

284. V každém uzlu stromu bude kromě celého čísla (tj. informační složky požadované v zadání úlohy) uložen jeden ukazatel, který bude ukazovat na pomocný seznam odkazů na následníky tohoto uzlu.

285. Použijte sinovou a kosinovou větu. Nezapomeňte uvést obě dvě řešení v případě, kdy je trojúhelník zadán dvěma stranami a úhlem proti kratší z nich.

289. Není výhodné počítat A^N jako součin N matic rovných dané matici A, lepší je počet potřebných násobení matic snížit. Postupným umocňováním na druhou vypočítejte matice A^1, A^2, A^4, A^8 atd., dokud exponent nepřekročí dané N. Výsledná matice pak bude součinem některých vhodně vybraných matic z této řady. Uvažte sami kterých. Pomůže vám zápis čísla N ve dvojkové soustavě.

290. Nejprve si předem spočítejte počet cifer největšího čísla, s nímž musíte umět pracovat, tj. čísla 50!. Tento počet cifer nemusíte stanovit přesně, stačí určit hodnotu, kterou počet cifer čísla 50! určitě nepřekročí. Číslo N!, které počítáte postupným násobením, si pak v programu reprezentujte v poli potřebné velikosti tak, že v každém prvku pole bude uložena jeho jedna cifra nebo raději skupina cifer (předem pevně zvolený počet cifer).

291. Nejprve si předem spočítejte počet cifer největšího čísla, s nímž musíte umět pracovat, tj. čísla 2^{100}. Tento počet cifer nemusíte

stanovit přesně, stačí určit hodnotu, kterou počet cifer čísla 2^{100} určitě nepřekročí. Číslo 2^N, které počítáte postupným násobením, si pak v programu reprezentujte v poli potřebné velikosti tak, že v každém prvku pole bude uložena jeho jedna cifra nebo raději skupina cifer (předem pevně zvolený počet cifer).

292. Znáte omezení maximálního možného počtu cifer v čísle, takže můžete čísla reprezentovat pomocí polí. V každém prvku pole bude uložena jedna nebo několik cifer (předem pevně zvolený počet cifer). Nezapomeňte, že díky přenosům do vyšších řádů může mít součet obou zadaných čísel 101 cifer.

293. Daná dlouhá čísla budou reprezentována spojovým seznamem svých cifer nebo skupin cifer. Pro potřeby operace násobení je výhodné, aby byl spojový seznam orientován tak, že na jeho začátku jsou cifry nejnižších řádů a na konci nejvyšších řádů. Násobte postupně odzadu jako při písemném násobení víceciferných čísel. Jednotlivé "řádky" vznikající násobením průběžně hned sčítejte.

294. Daná dlouhá čísla budou reprezentována spojovým seznamem svých cifer nebo skupin cifer. Pro potřeby operace dělení je třeba, aby byl spojový seznam orientován tak, že na jeho začátku jsou cifry nejvyšších řádů a na konci nejnižších řádů. Postupné odečítání dělitele od dělence, dokud je to možné, vede sice teoreticky ke správnému výsledku, ale může být velmi pomalé. Výrazně lepší je postupovat stejně, jako při písemném dělení víceciferných čísel na papíře.

295. Čísla postupně dělte jako při písemném dělení na papíře, postupně získávané cifry podílu ihned vypisujte a všechny hodnoty zbytků si ukládejte. Je-li v některém okamžiku zbytek nulový, byl již vypsán přesný výsledek dělení a celý výpočet proto ukončete. Jinak zkontrolujte, zda se nově získaný zbytek nerovná některému ze zbytků z minulých kroků výpočtu. Pokud ano, našli jste periodu v desetinném rozvoji. Perioda nemůže být delší, než jaká je hodnota dělitele.

297. Nejprve si spočítejte všechna prvočísla menší než N pomocí Eratosthenova síta.

298. Není dobré zkoumat systematicky všechny vybrané podposloupnosti a zjišťovat, která z nich je nejdelší rostoucí. Takový postup sice vede ke správnému výsledku, ale je velmi pomalý. Podstatného zrychlení výpočtu dosáhnete při použití pomocného pole celých čísel M délky 100. Postupně čtěte jednotlivé prvky zadané posloupnosti a přitom upravujte hodnoty uložené v poli M. Prvek M[J] bude v každém okamžiku výpočtu roven minimální dosud známé hodnotě posledního prvku vybrané rostoucí podposloupnosti délky J. Po přečtení všech čísel ze vstupu bude počet definovaných hodnot v poli M určovat výsledek, tj. délku nejdelší rostoucí vybrané podposloupnosti.

299. Je zbytečné vytvářet postupně celou popsanou posloupnost nebo třeba jen prvních K jejích členů. Úvahou o této posloupnosti si můžete ušetřit mnoho práce a určit přímo K-tý člen posloupnosti.

300. Je zbytečné vytvářet postupně celou popsanou posloupnost nebo prvních K jejích členů. Úvahou o této posloupnosti si můžete ušetřit mnoho práce a určit přímo K-tý člen posloupnosti.

301. Můžete procházet postupně všechna celá čísla a každé testovat, zda je požadovaného tvaru, dokud nenajdete prvních N čísel s touto vlastností. Takový postup je ale velmi nešikovný a pomalý. Mnohem lepší je vytvářet přímo hledanou posloupnost čísel tvaru $3^i \cdot 4^j \cdot 5^k$, a to dokonce hned v rostoucím pořadí. Využijte skutečnost, že každé číslo hledané posloupnosti je trojnásobkem, čtyřnásobkem nebo pětinásobkem některého z předchozích čísel nacházejících se v posloupnosti. Pro nalezení dalšího čísla požadovaného tvaru je ovšem třeba mít uložena buď všechna, nebo alespoň některá předchozí čísla.

305. Zkoušet postupně všechna možná rozdělení čísel do skupin je velmi pomalé a nešikovné. Pomůže předem provedená matematická úvaha.

307. Výraz postupně čtěte ze vstupu, čísla ukládejte do zásobníku a aritmetické operace ihned provádějte. Při přečtení znaménka aritmetické operace vezměte ze zásobníku vrchní dvě čísla, s nimi proveďte naznačenou operaci a výsledek opět uložte do

zásobníku. Po zpracování celého výrazu zůstane v zásobníku jediné číslo - výsledná hodnota výrazu.

309. Použijte čtyři vnořené cykly. Ve vnějším zkoumejte možné hodnoty x_5 z rozmezí od N/5 (největší z pěti sčítanců nemůže být menší!) do N-4. Pro každé takto zvolené x_5 nyní zbývá ještě rozložit $N-x_5$ mezi čtyři sčítance. V dalším cyklu proto procházejte všechny možné hodnoty sčítance x_4 z rozmezí od $(N-x_5)/4$ do $(N-x_5)-3$. Podobně postupujte i při zkoumání možných hodnot x_3 a x_2. Zbývající sčítanec x_1 pak již dopočítáte přímo jako $N-x_5-x_4-x_3-x_2$.

310. Použijte pět vnořených cyklů. Ve vnějším zkoumejte možné počty padesátikorun v zamýšlené platbě - jsou to všechny počty od 0 do N/50. Pro každý takový počet padesátikorun spočítejte, kolik ještě zbývá zaplatit, a potom podobným způsobem zkoušejte možné počty dvacetikorun, které se vejdou do zbývající částky. Podobně postupujte s platidly o hodnotách 10, 5 a 2 koruny. Po každém zvolení počtu platidel s hodnotami 50, 20, 10, 5 a 2 koruny se zbytek částky již jednoduše zaplatí pomocí jednokorun.

311. Rozklady můžete vytvářet v pomocném poli pomocí rekurzivní procedury. Abyste vyloučili opakování stejných rozkladů, které se liší pouze pořadím sčítanců, vytvářejte pouze takové rozklady, v nichž jsou hodnoty sčítanců (neostře) uspořádány. Procedura dostane informaci, kolik ještě zbývá rozložit a jak dlouhý (tj. kolikaprvkový) rozklad je již vytvořen a uložen v pomocném poli. Úlohu je možné řešit i bez použití rekurze. Musíte si jenom dobře zorganizovat práci a všechny rozklady vytvářet systematicky ve zvoleném pořadí.

312. Všechna přípustná rozmístění lidí postupně vytvářejte v pomocném poli pomocí rekurzivní procedury. Procedura může dostávat například informaci, kolik již přišlo lidí s desetikorunou a kolik s dvacetikorunou. Na základě těchto údajů zvolí přípustná další pokračování.

316. Nejprve si přesně definujte tvar římských čísel, s nímž budete v programu pracovat. Je třeba například zvolit, zda se bude číslo 99 zapisovat jako XCIX, či zda povolíte zápis IC.

317. Pro kódování z běžného textu do morseovky použijte pole indexované jednotlivými znaky. Hodnotami prvků pole budou řetězce znaků tvořící vždy příslušný kód v morseovce. Pro opačný směr kódování je lepší použít jinou datovou strukturu, neboť vyhledávání jednotlivých kódů morseovky v uvedeném poli by bylo zbytečně pracné. Jednou možností je například použít druhé pole, které obsahuje v každém svém prvku vždy dvojici tvořenou písmenem a jeho kódem v morseovce. Prvky tohoto pole jsou uspořádány vzestupně podle morseovkových kódů. Při kódování z morseovky do znaků pak můžete morseovkový kód vyhledávat v poli binárním vyhledáváním (půlením intervalů). Ještě lepší datovou strukturou je binární strom, v jehož uzlech jsou uložena písmena a číslice tak, že každá tečka nebo čárka v kódu morseovky znamená přechod na levého, resp. pravého následníka uzlu. Zpracováním celého morseovkového kódu jednoho znaku se tak dostanete od kořene stromu do uzlu, v němž je uloženo právě zakódované písmeno nebo číslice.

322. Přesná podoba výsledného programu bude záviset na tom, zda znáte tabulku znakových kódů, jakou používá váš počítač. Na osobních počítačích se obvykle pracuje s kódovou tabulkou ASCII. Její výhodou je, že kódy velkých písmen anglické abecedy v ní tvoří souvislý úsek uspořádaný podle abecedy od 'A' do 'Z'. Píšete-li program pro počítač používající kód ASCII, a nechcete-li tento program případně přenášet na jiný počítač, který by mohl pracovat s odlišnou kódovou tabulkou, můžete ve svém programu tuto vlastnost ASCII kódu výhodně využít. Pokud byste chtěli napsat program zcela univerzální a nezávislý na kódové tabulce počítače, bude kódování jednotlivých znaků trochu složitější. V takovém případě je možné použít například dvě pomocná jednorozměrná pole o 26 prvcích (to je počet písmen v anglické abecedě). Jedno bude indexováno celými čísly od 1 do 26 a jeho prvky budou znaky postupně od 'A' do 'Z', druhé pole bude naopak indexováno znaky z intervalu od 'A' do 'Z' a hodnotami jeho prvků budou celá čísla postupně od 1 do 26.

323. Viz návod k předchozí úloze.

324. Je vhodné, aby zadání mřížky a šifrovaný text byly uloženy v různých vstupních souborech.

325. Postupně čtěte slova ze vstupu a v pomocném znakovém poli si z nich vytvářejte "výstupní řádek". Slova v něm oddělujte zatím vždy jedinou mezerou. Když se již další přečtené slovo nevejde na právě vytvářený řádek, uschovejte si ho na začátek dalšího řádku. Vytvářený řádek ještě zformátujte rovnoměrným zvětšením mezer mezi slovy tak, aby poslední znak posledního slova končil na zadané pozici (šířka tisku) a řádek vytiskněte.

326. Dříve než začnete programovat, zvolte si přesný způsob zápisu všech povolených formátovacích příkazů a jejich význam. Příkaz na vložení tří volných řádků do textu může mít například tvar #V3.

336. Nezapomeňte, že klíčové slovo end v Pascalu slouží také k ukončení definice typu record a k uzavření příkazu case.

345. Na hledané kružnici budou jistě ležet alespoň dva ze zadaných bodů, neboť jinak by bylo možné kružnici zmenšit. Pokud na ní leží pouze dva body, musí být průměr kružnice roven jejich vzdálenosti (jinak by opět bylo možné kružnici obsahující všechny zadané body zmenšit). V opačném případě leží na hledané kružnici alespoň tři ze zadaných bodů. Je to tedy kružnice opsaná některému z trojúhelníků, jehož všechny tři vrcholy jsou mezi zadanými N body. Stačí tedy zkoumat všechny kružnice dvou typů: kružnice, jejichž průměrem je úsečka s krajními body v některé dvojici zadaných bodů, a kružnice opsané trojúhelníkům s vrcholy v daných bodech. Ze všech takových kružnic je třeba vybrat nejmenší takovou, která obsahuje všech N zadaných bodů.

346. Trojúhelník S leží celý uvnitř trojúhelníka T právě tehdy, jestliže uvnitř trojúhelníka T leží všechny tři vrcholy trojúhelníka S.

347. Aby bylo zadání úlohy srozumitelnější, vyjádříme ho ještě jednou jinými slovy. Úkolem je zjistit, zda existují čtyři celá čísla P,Q,R,S z rozmezí od 1 do N taková, že platí P<Q<R<S a současně $|A_PA_Q| = |A_QA_R| = |A_RA_S| = |A_SA_P|$.

348. Mnohoúhelník je konvexní právě tehdy, jestliže všechny jeho vnitřní úhly jsou menší než 180°.

350. Mezi danými N body určete ten, který má nejmenší y-ovou souřadnici. Je-li takových bodů se stejnou y-ovou souřadnicí více, vyberte z nich bod s největší x-ovou souřadnicí. Takto zvolený bod jistě patří do konvexního obalu a poslouží jako výchozí bod pro sestrojení celého konvexního obalu. Veďte úsečky z výchozího bodu do všech zbývajících bodů a ke každé z nich spočítejte úhel, který svírá s kladným směrem osy x. Porovnejte tyto úhly a vyberte úsečku, která leží na hranici konvexního obalu. Získali jste tím druhý bod hledaného obalu. Zbývající body již určíte jednotným postupem. Z bodu, který byl naposledy zařazen do konvexního obalu, veďte úsečky do všech ostatních bodů. Porovnejte úhly, které tyto úsečky svírají s dosud posledním úsekem vytvářeného konvexního obalu. Tím získáte další bod obalu. Tento postup opakujte tak dlouho, dokud se nevrátíte do výchozího bodu.

351. Matematickou indukcí lze snadno dokázat, že každý mnohoúhelník má nějakou triangulaci. Tuto skutečnost můžete využít při návrhu programu. Stačí nalézt jednu libovolnou úhlopříčku mnohoúhelníka (z uvedeného tvrzení o existenci triangulace vyplývá, že musí existovat), rozdělit pomocí ní mnohoúhelník na dva menší (tj. s menším počtem vrcholů) a s nimi postupovat dál stejně, dokud nezískáte samé trojúhelníky.

352. Jedná se o úlohu zjistit, zda dané dva vrcholy grafu (zde města) náležejí do téže komponenty souvislosti grafu (hranami grafu jsou zde přímé autobusové spoje). Úlohu můžete řešit tak, že vyjdete ze zadaného výchozího vrcholu a budete postupně označovat všechny vrcholy, které jsou z něho dostupné. Nejprve označíte vrcholy dostupné přímo z výchozího vrcholu, pro každý z takto označených vrcholů označíte další vrcholy, které jsou dostupné z něho, atd. Výpočet skončí, jestliže označíte cílový vrchol (je tedy dostupný) nebo když již nemůžete označit žádný další vrchol (označili jste již všechny dostupné, cílový vrchol mezi nimi není). Při programové realizaci uvedeného postupu je vhodné použít dvě pomocná pole velikosti N. První z nich je indexováno od 1 do N a je typu boolean - obsahuje informaci o každém vrcholu, zda byl označen. Druhé pole slouží k uložení čísel těch vrcholů, které již byly označeny, ale z nichž jste zatím "nerozšířili označení" na jejich dosud neoznačené sousedy.

353. Úkolem je nalézt jednu komponentu souvislosti daného grafu (tu, v níž leží vrchol grafu reprezentující výchozí město). Postupujte stejně jako při řešení předchozí úlohy. Výpočet ale ukončete teprve ve chvíli, když již nemůžete označit žádný další uzel.

354. Řečeno jazykem teorie grafů: Nalezněte všechny komponenty souvislosti daného grafu. Graf má N vrcholů (lidé přítomní na večírku), hrany grafu spojují dvojice lidí, o nichž je známo, že se narodili ve stejném roce. Při řešení postupujte obdobně jako v předchozí úloze, výpočet opakujte pro jednotlivé komponenty souvislosti.

356. Nejprve si spočítejte vzdálenosti mezi všemi dvojicemi letišť a vylučte takové dvojice, jejichž vzdálenost převyšuje dolet letadla. Ostatní vzdálenosti si uložte. Uložené vzdálenosti představují vlastně délky hran v grafu, jehož vrcholy reprezentují daná letiště. Dále již se vzdálenostmi letišť nemusíte pracovat. V sestrojeném grafu máte nalézt nejkratší cestu (z hlediska počtu hran) z výchozího do cílového vrcholu. K tomu můžete použít například postup prohledávání do šířky. Prohledávání začíná ve výchozím vrcholu. Postupně označte jedničkou všechny vrcholy přímo dostupné z výchozího vrcholu, dvojkou dosud neoznačené vrcholy, které jsou přímo dostupné z vrcholů dříve označených jedničkou atd., dokud nebude označen cílový vrchol. Jeho označení potom udává délku nejkratší cesty do něho vedoucí. Vlastní cestu potom určíte zpětným prohledáváním grafu od cílového vrcholu směrem k výchozímu. Přitom postupujte vždy po vrcholech, jejichž označení se liší o 1.

361. Je zbytečné počítat nezávisle na sobě jednotlivá kombinační čísla. Lepší je vytvářet Pascalův trojúhelník postupně po řadách a čísla v nově vytvářené řadě trojúhelníka spočítat vždy pomocí již známých čísel řady předchozí. Použijte k tomu součtový vzorec z kombinatoriky.

362. Nejprve si musíte spočítat, na který den v týdnu připadne 1. leden. Nezapomeňte počítat s přestupnými roky. Od roku 1582 se přestupné roky určují tak, že rok je přestupný tehdy, jestliže je jeho číslo dělitelné čtyřmi. Z tohoto základního pravidla ale existují výjimky v případě celých století. Je-li číslo roku dělitelné stem, je tento rok přestupný pouze tehdy, pokud je dělitelný také 400. Tedy např. roky 1700, 1800, 1900, 2100 nejsou přestupné, zatímco roky 1600 a 2000 jsou.

363. Nejprve spočítejte, kolik dní uplynulo ode dne narození člověka do začátku období, pro které vytváříte kondiciogram. Zbytky po celočíselném dělení 23, 28 a 33 pak udávají, v jaké fázi jednotlivých křivek je člověk na začátku zkoumaného období.

380. Postupujte metodou prohledávání do hloubky (prohledávání s návratem, backtracking). Šachovnici si v programu reprezentujte celočíselným polem. Na začátku práce programu budou v poli všude uloženy nuly. Postupně hledejte cestu koně po šachovnici a do jednotlivých polí šachovnice si přitom zapisujte čísla, která určují, kolikátým tahem vytvářené cesty se kůň na toto pole dostal. Při každém prodloužení zkoumané cesty zapište další číslo na některé pole, kde dosud byla nula. Naopak při návratu, pokud jste nedosáhli cíle a nemáte již další možnost tahu, čísla na šachovnici přepisujte opět nulami. Po dosažení cíle bude celá šachovnice zaplněna a čísla v ní uložená budou určovat průběh nalezené cesty šachového koně. Pamatujte na to, že pro některé rozměry šachovnice nemá úloha žádné řešení.

381. Úlohu můžete řešit systematickým zkoušením všech možných rozmístění dam na šachovnici. Využijte přitom metodu prohledávání s návratem (backtracking). Je dobré zavést si v programu pomocná pole, která budou průběžně evidovat, v kterém řádku, sloupci a úhlopříčce je momentálně umístěna nějaká dáma.

383. Při řešení úlohy můžete postupovat tzv. prohledáváním do šířky. Šachovnici si v programu reprezentujte celočíselným polem. Obsazená pole nejprve označte např. hodnotou -2, volná hodnotou -1 a výchozí pozici krále 0. Nyní postupně nahrazujte čísla -1 na šachovnici kladnými čísly udávajícími délku nejkratší cesty krále z jeho výchozího postavení na příslušné pole. Na volná sousední políčka výchozí pozice krále zapište číslo 1, potom na políčka sousedící s některým políčkem s číslem 1 (je-li tam zatím ještě -1, tzn. je tam volno a ještě jste nenašli kratší cestu na toto pole) napište číslo 2, atd. Postupujte tak dlouho, dokud nezapíšete nějaké kladné číslo na cílové pole (našli jste

nejkratší cestu krále a toto číslo udává její délku) nebo dokud nenastane stav, kdy již nemůžete nikam žádné další kladné číslo napsat (král se v takovém případě na cílové pole nemůže dostat). Při úspěšném ukončení uvedené fáze výpočtu můžete ještě vypsat cestu krále z výchozího na cílové pole. Postupujte zpětně od cílového pole k výchozímu a jděte vždy na některé ze sousedních polí s hodnotou o 1 menší. Získáte tak jednu možnou cestu krále minimální délky (může jich existovat více různých).

384. Postupujte prohledáváním do šířky podobně jako v předchozí úloze.

385. Obdobně jako při řešení předchozích dvou úloh můžete použít postup prohledávání do šířky.

386. Při řešení úlohy můžete použít například tzv. pravidlo pravé ruky. Pravou rukou se dotkněte stěny vpravo od vás a jděte bludištěm tak, abyste se pravou rukou stále dotýkali stěny. Pokud existuje cesta k cílovému otvoru ve vnější stěně, určitě k ní dojdete. Pokud taková cesta neexistuje, přijdete po chvíli zpět k místu, kde jste do bludiště vstoupili.

390. Nutnou podmínkou existence řešení je, aby počet všech čtverečků hrací plochy byl dělitelný pěti.

391. Nutnou podmínkou existence řešení je, aby počet všech čtverečků hrací plochy nebyl menší než pětinásobek celkového počtu hracích kamenů, které na ni máme umístit.

Nakladatelství **GRADA** pro Vás připravilo
v edici *EDUCA '99* tyto publikace:

Začínáme s PC - P. Rapant 144 str.

Kniha je názornou a základní učebnicí v oblasti užívání počítačů PC. Neklade proto žádné nároky na znalosti z oblasti výpočetní techniky. Popisuje technické vybavení PC - jeho obsluhu a údržbu. Věnuje pozornost práci s operačním systémem MS-DOS, obsahuje základní informace o ochraně souborů a o problematice počítačových virů. Obsahuje stručný přehled nejčastěji užívaných typů programových produktů. V publikaci je řada ilustrací a obrázků, které umožňují lepší přehled ve vykládané látce.

Zpracování textů na PC - Text602 - J. Horák 144 str.

Jedinečná kniha seznamuje studenty se psaním a úpravou textů na počítači řady PC. Úvod je věnován obecným informacím o práci s textem na PC a vysvětlení základních pojmů. Hlavní část knihy se zabývá českým textovým editorem Text602. Studenti zde naleznou efektivní a srozumitelný návod pro práci s kteroukoli dosavadní verzí tohoto editoru (až po 3.0), včetně celé řady řešených příkladů k procvičení.

Tabulkové procesory - Quattro Pro - F. Koschin 112 str.

Publikace je obecným úvodem do práce s libovolným tabulkovým procesorem, neboť všechny mají společnou "filozofii". V knize je vyloženo vše potřebné pro praktické řešení relativně jednoduchých problémů. Stěžejní část publikace je věnována výkladu práce s procesorem Quattro Pro. Kniha obsahuje řadu příkladů a praktických zkušeností autora, naučí každého základům práce a ovládání tohoto procesoru.

Databázové systémy - Paradox - Z. Havlíček 136 str. + disketa

Cílem této publikace je vysvětlit srozumitelnou formou základy relačního datového modelu a na praktických příkladech ukázat možnosti databázového zpracování dat; je určena studentům a všem uživatelům osobních počítačů, kteří začínají pracovat s databází. Teoretická část knihy vytváří předpoklady pro správné a cílevědomé využívání osobních počítačů při automatizaci běžných kartoték. Celý praktický výklad je založen na řešení příkladů pomocí databázového systému PARADOX verze 3.5, který umožňuje snadné a rychlé osvojení všech hlavních funkcí databázových systémů.

Myšlením k algoritmům - J. Kukal 136 str.

Kniha je určena především žákům středních škol. K jejímu studiu nejsou třeba žádné počáteční znalosti z oblasti výpočetní techniky a lze ji použít jako úvodní text před výukou konkrétních programovacích jazyků, její výhodou je metoda výkladu na příkladech. K výkladu algoritmů je důsledně použito jejich grafické znázornění pomocí strukturogramů.

Od problému k algoritmu a programu - I. Libicher, P. Töpfer 120 str.

Kniha je řešenou sbírkou zajímavých úloh z programování. Obsahuje úlohy, jejichž řešení vyžaduje hlubší zamyšlení na cestě ke kořenům problému a někdy i trochu matematické zběhlosti, přitom však neklade velké nároky na "technickou" programátorskou zručnost, a proto ji lze s úspěchem využít při výuce programování na všech typech škol.

Programátorský Babylón - kol. autorů 328 str. + disketa

Cílem publikace je poskytnout srovnání čtyř programovacích jazyků pomocí paralelně řešených úloh vPascalu, C++, assembleru a ve FoxPro. Většina úloh je řešena s dodržením jednotné filozofie a jednotného označení pojmů, takže čtenář ovládající alespoň jeden z uvedených jazyků se rychle zorientuje. Obtížnost úloh se postupně zvyšuje.

Začínáme s programováním - R. Kryl, J. Drózd 312 str.

Vladimírem Jiránkem ilustrovaná kniha je čtivě napsanou základní učebnicí programování postavenou na programovacím jazyce Pascal. Autoři se snaží představit programování jako součást lidské kultury, nikoli jako souhrn příkazů a triků. Velký důraz je věnován pochopení pojmu algoritmus a osvojení algoritmického stylu myšlení. Programovací jazyk je sice důležitým, ale přesto jen nástrojem pro dosažení cíle. Velké množství příkladů vypracovaných až do fáze programu umožňuje použít knihu i samoukovi.

Prostorové modelování - J. Pelikán 144 str.

Kniha je učebnicí prostorového modelování se zaměřením na využití počítačů řady PC v této problematice. Popisuje a rozebírá různé metody konstrukce obecných ploch od těch nejjednodušších po složitější a moderní postupy. Text je doplněn velkým množstvím ilustrací názorně doplňujících probíranou látku.