

Tady je tedy slíbený seznam různých problémů na témata probíraná tento semestr. **Deadline** na odevzdání je do posledního cvičení **11.1.2017**. Nejlepší řešitel dostane čokoládu. Ještě připomínám, že součástí řešení je kromě jeho popisu také **zdůvodnění správnosti, odhad časové a prostorové složitosti a jejich zdůvodnění**.

1. Dlaždice (8 bodů):

Vstup Množina barev, množina typů dlaždic (od každého typu máte neomezené množství dlaždic), ohodnocení krajů koupelny.

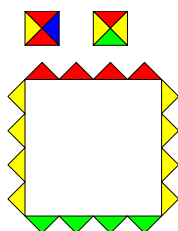
Popis problému Úkolem je vydláždít obdélníkovou koupelnu.

Dlaždicemi nemůžeme otáčet. Každá dlaždice má barvu na horní, dolní, levé i pravé části (barvy můžou a nemusí být stejné). Když dláždíme, tak musí barvy navazovat na sebe (tj. pokud jsou dvě dlaždice vedle sebe, tak pravá strana první musí mít stejnou barvu, jako levá strana té druhé).

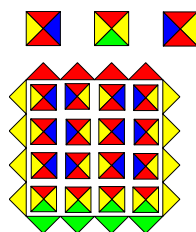
Obdobně koupelna má na všech čtyřech krajích barvy, které musí sedět s barvami použitých dlaždic.

Výstup Jde danou plochu vyplnit dlaždicemi, tak aby na sebe barevně navazovaly a zároveň barevně navazovaly na okraje?

Příklad Příklad koupelny, která nelze vydláždít a pod ní příklad koupelny, která lze vydláždít.



Obrázek 1: Koupelna, která nelze vydláždít.



Obrázek 2: Koupelna, která lze vydláždít.

2. Hra (10 bodů):

Vstup Množiny matematiků A , B ($A \cap B = \emptyset$), množina dvojic, kdo s kým napsal článek.

Výstup Kdo má vyhrávající strategii? Popřípadě jakou?

Popis hry Hráči se střídají v tazích. Hráč jedna vybírá matematika $a_i \in A$ a hráč 2 vybírá matematika $b_i \in B$. Platí b_i napsal článek s a_i ; a_{i+1} napsal článek s b_i . Hráč prohrává pokud nemůže vybrat následujícího matematika, aniž by porušil podmínku.

Příklad vstupu $A = \{X, Y, Z\}$, $B = \{\alpha, \beta, \gamma, \delta\}$, články $\{(X, \alpha), (X, \delta), (Y, \beta), (Y, \gamma), (Z, \alpha), (Z, \beta), (Z, \gamma)\}$

Příklad výstupu Vyhraje hráč 2.

3. Královská jména (7 bodů):

Popis problému V jednom království král rozhodl, že platné jméno je pouze takové, které má minimálně n_0 a maximálně n_1 písmen a neobsahuje jako podslovo jeho jméno. Vaším úkolem je spočítat, kolik jmen mají jeho poddaní na výběr.

Vstup Jméno krále K , čísla n_0, n_1 a m písmen abecedy užívané v království (ve výsledku se písmena mohou opakovat).

Výstup Počet řetězců délky n , kde $n_0 \leq n \leq n_1$, které neobsahují K jako podslovo.

Příklad vstupu Abeceda $\{a, b, c, d\}$, zakázané jméno bad , slova délky 3-4.

Příklad výstupu 311

4. Tučňáci (7 bodů):

Popis problému Tučňáci stojí na začátku každý na své kře a chtějí se potkat všichni najednou na jedné kře. Každý tučňák může skočit na libovolnou kru ve vzdálenosti nejvýše d . Každým doskokem tučňáka se kra trochu poničí. Naštěstí víme, kolikrát se na kterou kru dá skočit, než se zcela rozpadne. Otázkou je, na kterých krách se mohou tučňáci potkat.

Vstup Množina ker (každá popsána umístěním a výdrží), množina tučňáků (každý na nějaké kře), číslo d .

Výstup Kry, na kterých se mohou tučňáci potkat.

Příklad vstupu Délka jednoho tučňáččeho skoku $d = 1$; kra na pozici (1,0) s tučňákem a výdrží 1; kra na pozici (2,0) s tučňákem a výdrží 1; kra na pozici (3,0) s výdrží 1.

Příklad výstupu Mohou se potkat na první i druhé kře.

5. Známky (12 bodů):

Popis problému Ve škole jste napsali k testů, u každého šlo získat maximálně M_i bodů a vy jste získali Z_i z nich. Bude Vám škrtnuto ℓ nejhorších testů (s nejhorším procentuálním výsledkem Z_i/M_i). Zjistěte, pro která ℓ lze dosáhnout lepšího výsledku škrtnutím jiných testů. Výsledkem je součet získaných bodů z neškrtnutých testů lomeno součet všech bodů z neškrtnutých testů.

Vstup Výsledky testů a maximální počty bodů, které šlo získat.

Výstup Vypište hodnoty ℓ , pro která lze dosáhnout lepšího výsledku, když neškrtneme ℓ procentuálně nejhorších testů.

Příklad vstupu (1, 2), (5, 9), (3, 8), (4, 10), (1, 3)

Příklad výstupu 1, 2

6. Rozhodovací stromy (5 bodů):

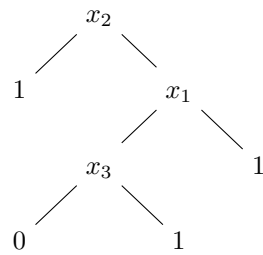
Popis problému Máme daný binární strom, kde jeho vnitřní vrcholy jsou indexované pomocí proměnných x_i a levý podstrom odpovídá nastavení $x_i = 0$ a pravý $x_i = 1$. V listech jsou hodnoty 1, 0. Pomocí takového stromu tedy máme danou nějakou funkci $f(x_1, \dots, x_n)$, jejíž výsledek zjistíme tak, že jdeme z kořene stromu vždy do podstromu označeného hodnotou příslušné proměnné a v listu přečteme výsledek.

Máme dané dvě funkce (pomocí jejich stromů). Cílem je zjistit, jestli je jedna podfunkcí druhé (tj. jestli je její strom podstromem druhé).

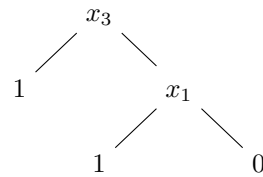
Vstup Dva rozhodovací stromy

Výstup ANO/NE

Příklad vstupu



Tato funkce tedy odpovídá $f(0, 0, 0) = f(0, 0, 1) = f(1, 0, 0) = f(1, 0, 1) = f(1, 1, 0) = f(1, 1, 1) = f(0, 1, 1) = 1$ a $f(0, 1, 0) = 0$.

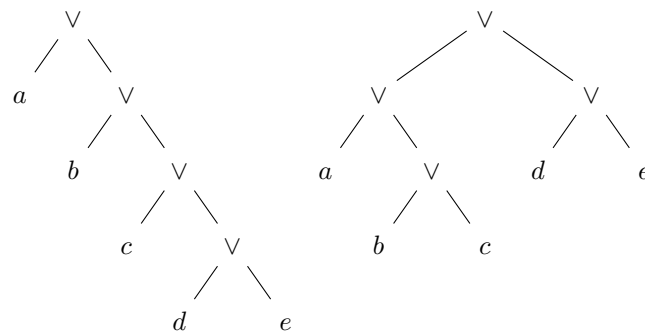


Příklad výstupu NE

7. Formule (7 bodů):

Popis problému Popište jak pro formuli vytvoříte formuli minimální hloubky. Jaká bude její hloubka? Ve formuli se mohou objevit pouze binární \vee, \wedge a unární \neg .

Například formule $a \vee (b \vee (c \vee (d \vee e)))$ má hloubku 5, ale ta samá formule napsaná jako $(a \vee (b \vee c)) \vee (d \vee e)$ má hloubku 4.

**8. Nejdelší podslovo (12 bodů):**

Popis problému Definujeme podobnost k řetězců, jako nejdelší úsek, který mají všechny řetězce společný.

Vstup k řetězců (dvě varianty - $k = 2$ je dané předem, k je součástí vstupu)

Výstup Číslo P určující podobnost vstupu.

Příklad vstupu První varianta: řetězce "zvukomalebny" a "omalovánka"

Druhá varianta: $k = 3$ "střípek", "skřítek", "skřípot"

Příklad výstupu První varianta: 4

Druhá varianta: 2

9. Nepozorný student (4 body + [0.1, 0.5] za chybu):

Opravte úkol, který Vám student odevzdal. Bohužel pracoval na něm před cvikem dlouho do noci, po tom co se vrátil z flámu se spolužáky. Zkuste přijít na to, který problém (z témat probíraných na přednášce) se snažil řešit a opravit alespoň část chyb.

```
1 while True do
2   pom = 0; pom2 = n; Y = False;
3   for (i, i ≤ n, ++i) do
4     for (j, j ≤ n, ++j) do
5       if i.weight() ≤ j.weight() then
6         X[i] = X[j];
7         X[j] = X[i];
8       end
9     end
10  end
11  for i = 0, i ≤ 5, ++i do
12    if Y[pom] == False then
13      q = X[pom].v[0].neighbours();
14      while True do
15        for i ∈ q do
16          q.add(i.neighbours());
17          if i == X[pom].vertex[2] then
18            c = True;
19          end
20        end
21      end
22    end
23    pom --;
24    if c! = True then
25      Y[pom] = True;
26    end
27  end
28  for i = 0, i ≤ 5, ++i do
29    for v ∈ X[pom2].neighbours() do
30      if X[v, pom2].weight < actual_w then
31        pom = v;
32        actual_w = X[v, pom2].weight
33      end
34    end
35    strom[v].join(pom2);
36    Y[pom, pom2] = True;
37    pom2 --;
38  end
39 end
```