Informatika – navazující magisterské studium
Přijímací zkouška z informatiky – 2012 – varianta A

Každá úloha je hodnocena maximálně 25 body.
Všechny své odpovědi zdůvodněte!

1. Kolika způsoby můžeme vynechat tři čísla z posloupnosti

$$1\ 4\ 2\ 5\ 7\ 3\ 8\ 9,$$

abychom získali rostoucí posloupnost?

2. Převeďte do disjunktivní normální formy následující logickou formuli:

$$((u \ \&\ y) \vee (\sim u \ \&\ \sim y)) \ \&\ (\sim (x \vee z)) \ \&\ (\sim z)$$

Znaky x, y, z, u označují logické proměnné, znak & představuje logickou spojku konjunkce, znak $\vee$ označuje disjunkci, symbolem $\sim x$ značíme negaci proměnné x.

3. Navrhněte deterministický konečný automat nad abecedou {0, 1}, který přijímá všechna slova délky aspoň 2 znaky, která obsahují sudý počet znaků 0 a lichý počet znaků 1. Například slova 001, 11111, 0101010 automat přijme, zatímco slova 1, 0000, 100101 nepřijme. Přechodovou funkci automatu zapište ve tvaru tabulky a automat znázorněte ve tvaru přechodového diagramu. Navrhněte co nejjednodušší automat, tzn. takový, který bude mít co nejméně stavů.

4. Je dán následující program (obě zadání v Pascalu a v jazyce C jsou ekvivalentní):

```
program AAA;
var A, B, C: integer;
begin
read(A, B);
while A > B do
  A := A – 2;
while B > A do
  B := B – 2;
C := A + B;
writeln(C)
end.
```

```
main()  /* AAA */
{
  int a, b, c;
  scanf("%d %d",&a,&b);
  while (a > b)
    a -= 2;
  while (b > a)
    b -= 2;
  c = a + b;
  printf("%d", c);
}
```

Určete, jak závisí výsledná hodnota proměnné C (tzn. výstup programu) na vstupních hodnotách proměnných A, B.

Řešení přijímací zkoušky z informatiky – 2012 – varianta A

1. Musíme vyřešit poklesy v posloupnosti mezi čísly 4 – 2 a 7 – 3, všude jinde zadaná posloupnost roste. První z uvedených poklesů vyřešíme buď odebráním čísla 4, nebo odebráním čísla 2. Druhý pokles lze vyřešit buď odebráním dvojice čísel 5 7, nebo odebráním čísla 3. První možnost (odebrání dvojice 5 7) lze použít jen tehdy, když jsme první pokles řešili odebráním čísla 4. Varianta s odebráním čísla 3 může následovat po předchozím odebrání jak 2, tak i 4. V obou případech je pak potřeba odebrat ještě jedno třetí číslo, na jehož volbě nezáleží. Buď s číslem 3 odebereme 4 i 2 zároveň, nebo jen jedno z nich a k tomu ještě některé z pěti zbývajících čísel. Celkem tudíž existuje 1 + 1 + 2 . 5 = **12** řešení.

2. Můžeme buď provádět postupné úpravy zadané logické formule až do požadovaného tvaru disjunktivní normální formy, nebo využít Karnaughovy mapy, nebo například sestavit tabulku logických hodnot:

x	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
y	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
z	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
u	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
((*u* & *y*) ∨ (~*u* & ~*y*)) & (~(*x* ∨ *z*)) & (~z)	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0

Formuli v disjunktivní normální formě získáme jako disjunkci těch sloupců tabulky, v nichž nabývá zadaná formule pravdivostní hodnoty 1:

(~*x* & ~*y* & ~z & ~*u*) ∨ (~*x* & *y* & ~z & *u*)

3. K řešení úlohy stačí konečný automat se šesti vnitřními stavy. Pomocí stavů automatu rozlišíme, zda již byly přečteny ze vstupu alespoň dva znaky, a pokud ano, zda byl dosud přečten sudý nebo lichý počet znaků 0 a 1 (čtyři kombinace – stavy D, E, F, G). Z nich koncový bude pouze stav G, který odpovídá sudému počtu znaků 0 a lichému počtu znaků 1. Stavy B a F jsou ekvivalentní, takže stav B můžeme vynechat. Minimalitu počtu stavů poté ověříme redukcí sestrojeného konečného automatu.

	0	1	
→ A	B	C	A = vstupní stav, žádný znak nebyl přečten
B	D	E	B = přečten jeden znak 0
C	E	D	C = přečten jeden znak 1
D	F	G	D = sudý počet 0, sudý počet 1
E	G	F	E = lichý počet 0, lichý počet 1
F	D	E	F = lichý počet 0, sudý počet 1
← G	E	D	G = sudý počet 0, lichý počet 1

4. Vstupní hodnoty proměnných A, B si označíme A_0, B_0. Rozlišíme tři případy:
a) Mají-li vstupní hodnoty A_0, B_0 stejnou paritu, provede se nejvýše jeden z cyklů v programu. Proběhne podle potřeby tolikrát, až se proměnná s vyšší vstupní hodnotu bude přesně rovnat proměnné s nižší vstupní hodnotou. Obě proměnné tak dosáhnou hodnoty min(A_0, B_0) a výsledná hodnota proměnné C bude rovna **2 min(A_0, B_0)**.
b) Mají-li vstupní hodnoty A_0, B_0 různou paritu a přitom $A_0 < B_0$, první z cyklů v programu se neprovede ani jednou a ve druhém se potom proměnná B snižuje tak dlouho, až dosáhne hodnoty A_0-1. Výsledná hodnota proměnné C bude proto rovna **2 A_0 - 1**.
c) Mají-li vstupní hodnoty A_0, B_0 různou paritu a přitom $A_0 > B_0$, bude se v prvním cyklu proměnná A snižovat tak dlouho, až dosáhne hodnoty B_0-1. Tím ale výpočet nekončí. Poté je totiž splněna podmínka druhého cyklu a jeho tělo bude provedeno přesně jednou, přičemž se hodnota proměnné B sníží o 2 na hodnotu B_0-2. Výsledná hodnota proměnné C bude proto rovna **2 B_0 - 3**.