Podzim 2007, 20.11.2007, Vnitro
Příklad 01

Priklad 1 (max 10b): Mali ste definovat nedeterministicky konecny automat a jeho rozsirenu prechodovu funkciu.

Kedze islo o definicu, tak sa podla toho odvija aj bodovanie a pripadne zrazky.
a) definicia NFA - 6 bodov:
1 bod za to, ze je to patica (Q, \Sigma, \delta, q0, F)
po jednom bode ste dostali za vysvetlenie kazdej jej polozky:
- Q - neprazdna konecna mnozina stavov
- \Sigma - konecna vstupna abeceda
- \delta - prechodova funkcia Q x \Sigma -> 2^Q
- q0 - pociatocky stav, q0 \in Q
- F - mnozina akceptujucich stavov, F \subseteq Q
b) rozsirena prechodova funkcia - 4body:
1 bod za typ funkcie \delta': Q x \Sigma^* -> 2^Q
1 bod za \delta'(q, \eps) = {q}
2 body za \delta'(q, wa) = velke zjednotenie ...

Mnohi z vas zabudali uviest, ze sa jedna o konecnu mnozinu stavov a konecnu abecedu znakov. V pripade, ze vam slovko "konecna" chybalo pri oboch mnozinach, tak ste prisli o 1 bod.
Specialne tiez nestacilo iba uviest, ze \delta je prechodova funkcia. Bol potrebny aj jej typ Q x \Sigma -> 2^Q, kedze iba tak poznat, ci popisujete NFA alebo DFA. Tento bod ste mohli ziskat aj ked ste mali slovne popisane, ze NFA (narozdiel od DFA) umoznuje prechod z jedneho stavu pod jednym znakom abecedy do viacerych roznych stavov.
Casto ste popisovali rozsirenu prechodovu funkciu pre DFA. Obecne som sa snazil aspon cast z toho obodovat. Vyslo to tak, ze za spravnu definiciu NFA a rozsirenej prechodovej funkcie pre DFA ste mohli ziskat maximalne 7 bodov, ak bol aj automat deterministicky, tak 6.

Caste chyby:
Okrem toho, ze mnohi nemaju potuchy, co znamena, ze automat je deterministicky (zamienanie za deterministicky resp. automat s \eps krokmi patrilo k tym mensim chybam), tak ste casto plietli gramatiky a automaty dokopy. Takze napr. Q bolo mnozinou neterminalov alebo dokonca vstupnym/generovanym jazykom. A \delta bola mnozina pravidiel alebo predpisov prechodovych funkcii. Dalsou castou chybou bolo, ze ste q0 oznacili za mnozinu vstupnuch stavov, respektive F bol iba jeden akceptujuci stav.

Podzim 2007, 20.11.2007, Vnitro
Příklad 02

Priklad 2 (max 8b): Kolko slov ma jazyk L* \ (L+)^2, kde L = {aa, ab, ba, bb}
8 bodov ste ziskali za spravnu odpoved 5 slov (stacilo aj samotne cislo, bez uvedenia vysledneho jazyka).
2 body ste ziskali za odpoved, ze vysledy jazyk obsahuje iba \eps.
Rovnako, o 2 body ste prisli, ak vam vo vyslednom (a inak spravnom) jazyku \eps chybalo.
Za kazdu dalsiu spravnu vec som sa vam snazil dat po bode. (napr. definica L a pod.) Smolu mali ti, co napisali iba vysledok. Tym uz nebolo co obodovat.*
Naopak, ak ste sice mali spravnu odpoved, ale uviedli ste tam aj nejake vylozene nepravdive tvrdenie (napr. ze L je konecny jazyk), tak ste o 1 bod prisli.*

Najcastejsou chybou bolo, ze ste (L+)^2 nespravne interpretovali ako:
(L^1)^2 \zjednotene (L^2)^2 \zjed (L^3)^2 \zjed ...
Tym padom ste v (L+)^2 uvazovali iba slova s dlzkou 4k, k>0. A vysledny jazyk mal nekonecne vela slov.

Dalsimi castymi chybami boli napriklad tvrdenia typu:
L, L+ su konecne jazyky. L+ je sice nekonecny, ale (L+)^2 je konecny.*
(L+)^2 = (L^2)+
L+ = (L+)^2
L = L^0 . L^1 . L^2 . ...*
a^2b^2 = abab

Podzim 2007, 20.11.2007, Vnitro
Příklad 03, 04

Příklad 3 (max 10 bodů):

Správné řešení byl jakýkoliv automat jehož stavy se daly označit čisly z množiny {0, 1, 2, 3, 4} tak, aby z každého stavu označeného i směřoval přechod pod písm. a do nějakého stavu označeného (i + 1) mod 5 a přechod pod písm. b do stavu označeného (i + 2) mod 5. Počáteční stav byl kterýkoliv označený 0 a akceptující byly všechny stavy označené 1.
Z toho vyplývá, že min. automat má 5 stavů a jeho pěknou grafickou reprezentaci uveřejnil pan Dvořák ve vlákně "vnitrosemestralni pisemka". Rekord v počtu stavů správného řešení na písemce byl tuším 36 :).

Teď bodování:
Automat, který akceptuje nějaký konečný jazyk, který je podmnožinou L - 1 bod
Automat, který akceptuje nějaký nekonečný jazyk, který je ostrou podmnožinou L - 2 body
Automat, ze kterého byla vidět určitá myšlenka ("skákání" pod a o 1 stav, pod b o 2 stavy) - až 5 bodů
Víc než 5 bodů měl ten, kdo to měl "v podstatě správně", jenom s určitými méně či více závažnými chybami (chybějící přechody, špatně nasměrované některé přechody, neoznačené akceptující stavy, počítání modulo 6 místo 5 atd...).
Za jeden chybějící přechod jsem nechával ještě plný počet bodů, za dva chybějící jsem strhával 1 nebo 2 body, podle toho, které chyběly. Obdobné srážky byly za ostatní prohřešky.

Příklad 4 (max. 10 bodů):
a) (max 5 bodů):
L = L\R = L\(L průnik R) = L\(prázdná množina) = L, tedy implikace platí.
Za správnou odpověd bez zdůvodnění 2 body. Za vyloženě špatné zdůvodnění jste o jeden z nich přišli. Částečně správná zdůvodnění se prakticky nevyskytla.
b) (max 5 bodů):
(prázdná množina) sjednoceno (libovolný neregulární jazyk) je neregulární jazyk, a protože prázdná množina je regulární, implikace neplatí.
Za správnou odpověd bez zdůvodnění 2 body. Za vyloženě špatné zdůvodnění jste o jeden z nich přišli. Částečně správná zdůvodnění se prakticky nevyskytla.

Podzim 2007, 20.11.2007, Vnitro
Příklad 05

Rozhodněte a posléze dokažte, zda je jazyk L={a^j.b^k | j sudé či j=k} regulární.

Jazyk L není regulární.

Vzorové důkazy:
Pomocí PL: Pro libovolné přirozené číslo n uvážíme slovo w=a^(2n+1).b^(2n+1). Každé rozdělení w=xyz tvaru x=a^k, y=a^l, z=a^(2n+1-k-l).b^(2n+1), kde l>0 a k+l<=n pak můžeme napumpovat na xy^3z, což je slovo a^(2n+1+2l).b^(2n+1), které ovšem není v L, neboť 2n+1+2l je liché a různé od 2n+1. Dle PL tedy L není regulární.

MN větou: Dokážeme, že index ~L je nekonečný. Stačí uvážit nekonečnou množinu slov M={a^(2i+1) | i přirozené}. Pro libovolná dvě slova a^(2i+1), a^(2j+1) uvážíme příponu b^(2i+1). Zřejmě a^(2i+1).b^(2i+1) leží v L, ale a^(2j+1).b^(2i+1) nikoli. Proto neplatí a^(2i+1) ~L a^(2j+1). Jelikož slova jsme vzali libovolná dvě z M, patří každé slovo z M do jiné třídy rozkladu Sigma*/~L, tedy index ~L je nekonečný. Z MN věty plyne neregularita L.

Uzávěrovými vlastnostmi:
Víte ze skript, že reg. jazyky jsou uzavřeny na inverzní homomorfismus. Dále víte, že jazyk X={a^n.b^n | n nezáporné celé} není regulární (opět ze skript). Uvažme homomorfismus f, který a zobrazí na aa, b na bb. Kdyby f(X)={a^2n.b^2n | n nezáporné celé} byl regulární, tak by ale i X musel být regulární, tedy f(X) není regulární.
Dále f(X) = {a}.(L\{aa}*.{b}*).{b} u {epsilon} a kdyby L byl regulární, tak f(X) je regulární, což ale není. Tedy L není regulární.

Bodování:
První tři body jste mohli získat za to, že jste splnili první část zadání: odpověděli jste správně na otázku, zda je L regulární.

Dalších 12 bodů jste mohli získat za důkaz neregularity. Nejčastější bylo použití PL, ovšem v drtivé většině si řešitelé zvolili nevhodné slovo a^n.b^n, které jde vždy rozdělit tak, aby při jakémkoli napumpování zůstalo v jazyce. U těch, kteří použili (Myhill-)Nerodovu větu se zase často stávalo, že o nějaké nekonečné množině slov nedokázali, že žádná dvě nejsou v relaci ~L, ale pouze některé vybrané dvojice (někdy třeba jen konečně mnoho takových dvojic).
Ani jeden důkaz založený pouze na uzávěrových vlastnostech nebyl dobře. Vzpomeňte si na kapitolu 5 ze sbírky. Dělali jsme na cvičení 6 příkladů v nichž jste dokazovali neplatnost řady tvrzení, která pak mnozí z vás prohlásili za platná.

K bodování důkazu jsem přistupoval takto: 12 bodů jsem rozdělil na 7 za strukturu a 5 za hodnoty. Napsal-li někdo dobře, jak by se použilo např. PL, ale vybral špatné slovo w a další parametry, dostal 7 za strukturu důkazu, ale třeba jen 2 za hodnoty. Často jsem toto rozdělení do řešení přímo napsal označené písmeny S a H.
Mnozí z vás dokazovali, že jazyk {a^n.b^n | n nezáporné celé} není regulární. To sice nebylo vaším úkolem a nikdo z vás to nebyl schopen použít k důkazu neregularity L, ale pokud ten důkaz byl dobře napsaný, nějaké body jste také dostali. Byly pak označeny slovy "špatný jazyk".

Podzim 2007, 20.11.2007, Vnitro
Příklad 06

Priklad 6 spocival v prevodu regularniho vyrazu na konecny automat pomoci algoritmu. Plny pocet byl 12 bodu.

Vase hodnoceni probihalo podle nasledujiciho bodovani:

*3 za prevody E*F*
2 za prevody E+F
1 za prevody E.F
2 za spravne poradi provedeni jednotlivych kroku -- znalost priority operaci, respektovani zavorek,...
1 za spravne odstraneni hrany s \emptyset (regularnim vyrazem odpovidajicim prazdne mnozine)
3 za celkovou spravnost (selsky rozum, spravnost po formalni strance, zda automat opravdu akceptuje jazyk popsany regularnim vyrazem, zda se jedna o konecny automat,...)

Vase typicka reseni, caste chyby a jejich bodovani:
- Spravne reseni podle algoritmu 12b.
- Chyba: regularni vyraz \emptyset byl odstranen intuitivne nekde v prubehu reseni, proste se vyparil, podle algoritmu se zpracovava odstranenim _hrany_ s \emptyset. Pokud bylo Vase reseni jinak spravne dostali jste take 12b. (ale priste hodnoceni nemusi byt tak tolerantni)
- Chyba: hrana ohodnocena regularnim vyrazem \emptyset byla nahrazena hranou s \epsilon. (-1b).
- Chyba: na konci zustala v KA hrana ohodnocena \emptyset -- nejedna se tedy o KA (-2b).
*- Chyba: pri zpracovavani E*F zapominate epsilon kroky - za kazde spravne zpracovani +1b (ve vyraze se celkem vyskytovaly 3)*
- Chyba: spatne poradi operaci (casto napred zpracovani ., pak +) (-3b).
- Chyba: nerespektovani zavorek (-2b)
- Chyba: vubec nepostupujete podle algoritmu, ale dojdete k automatu, ktery akceptuje dany jazyk, dostavate 2b.
- Chyba: nepostupujete podle algoritmu, operace ktere provadite jsou nekdy i logicky spravne, nekdy ne - bodovano individualne podle zavaznosti nedodrzeni algoritmu a vlivu na jazyk akceptovany automatem.

Podzim 2007, 20.11.2007, Vnitro
Příklad 07

Je dana abeceda Sigma = {a,b}

a)Rozhodnete zda existuje konecny jazyk jehoz ~L ma index 3
Odpoved: existuje napr L={a} max 5 bodu
uvedeni jazyka jehoz minimalni automat(zduraznuji ze musi byt deterministicky a uplny) ma 3 stavy - 5 bodu
pokud minimalni automat pro vas jazyk mel 2 nebo 4 stavy obdrzeli jste 3 body (velice casto jste zapominali na uplnost)
pokud minimalni automat pro vas jazyk mel jiny pocet stavu obdrzeli jste 2 body
pokud vas jazyk nebyl konecny obdrzeli jste 1 bod
za odpoved ano jste obdrzeli 1 bod

b)Rozhodnete zda existuje nekonecny jazyk jehoz ~L ma index 3
Odpoved existuje napr L={a^+} max 5 bodu
uvedeni jazyka jehoz minimalni automat ma 3 stavy - 5 bodu
pokud minimalni automat pro vas jazyk mel 2 nebo 4 stavy obdrzeli jste 3 body
pokud minimalni automat pro vas jazyk mel jiny pocet stavu obdrzeli jste 2 body
pokud vas jazyk byl konecny nebo nebyl regularni obdrzeli jste 1 bod
za odpoved ano jste obdrzeli 1 bod

c)Rozhodnete zda existuje konecny jazyk jehoz ~L ma index nekonecno
Odpoved neexistuje
Mozny (nejjednodusi) dukaz:
L je konecny => L je regularni =>(M.N veta) index ~L je konecny
za uvedni formalniho (korektniho) dukazu 5 bodu
Jelikoz hodne z vas(k nasemu prekvapeni) melo s timto prikladem problemy snazil jsem se ho hodnotit co nejmirneji

pokud jste zapomeli argumentovat ze plati L je konecny => L je regularni a jinak byl dukaz v poradku obdrzeli jste 4 body
pokud vas dukaz mel dobrou myslenku(nebo aspon naznak dobre myslenky) ale nebyl formalne presny obdrzeli jste 2-4 body
za odpoved ne jste obdrzeli 1 bodu

Caste a bohuzel spatne argumentace byly typu:
1)Konecny jazyk ma jen konecne mnoho trid rozkladu.
Tato argumentace je naprosto spatna jelikoz jazyk nema zadne tridy rozkladu ale je tvoren sjednocenim nejakych trid rozkladu mnoziny Sigma^* podle relace ~L (tato relace je dana prave jazykem L).
2)Pokud je jazyk konecny pak index relace ~L je taky konecny jelikoz v nejhorsim pripade je kazde slovo v jedne tride rozkladu podle ~L.
Tato argumentace ja taky spatna protoze nerika nic o slovech ktere nepatri do jazyka(a techto slov je nekonecne mnoho)
3)U konecneho jazyka L je index ~L <= |L|
Toto neplati, uvazme jazyk nad abecedou Sigma L={a^100} pak |~L| = 102 > |L| =1
Pokud jste argumentovali pouze temito argumety obdrzeli jste 1 bod

Take se objevilo par dukazu ktere nevyuzivaly M.N vetu a byly temer korektni a proto byly ohodnoceny 4-5 body.

Tento vycet spatnych nebo nedostatecnych argumentu neni konecny ale doufam ze jsme zde uvedl reprezentanty tech nejcastejsich a ze vam to pomuze k pochopeni nejen tohoto prikladu ale take k pochopeni regularnich jazyku.

Na zaver toho prispevku bych rad uvedl poznamku k nekterym vasim resenim. Jeden z cilu toho predmetu je naucit studenty formalnimu vyjadrovani jak v ustnim tak v pisemnem projevu. Bohuzel nektere zapisy jazyku byly tak nepresne ze jsem si mohl jen domyslet jaky jazyk jste meli na mysli. Proto na vas apeluji aby jste se v pristich pisemnych projevech snazili o presne zapsani vasich reseni coz je urcite take ve vasem zajmu.

Podzim 2008, 20.11.2008, Vnitro
Příklad 01, 02

1. příklad (max. 10 bodů):

Za definici množiny regulárních výrazů bylo 5 bodů, za definici L(E) zbylých 5. Tyto dvě věci musely být v řešení řádně odděleny. Kdo například uvedl pouze definici L(E), dostal jen 5 bodů. Argument, že definice regulárního výrazu se z toho dá "vyčíst" není dostatečný.

Ten kdo v některé části zapomněl na nějaké položky, ztratil v té části 1-2 body, ale pouze pokud nešlo o nic závažného. Jinak byly srážky výraznější. Kdo například v první části úplně zapomněl na zákl. regulární výrazy a napsal jen, jak se tvoří složitějsí, dostal jen 1 bod.

2. příklad (max. 8 bodů):

Správné řešení:

L = {a, b, aa, ab, ba, bb}
L^* = {a, b}^*
co-L = {\epsilon} U {a, b}^3{a, b}^*
(co-L)^2 = co-L
L^*\(co-L)^2 = L

Výsledný jazyk se tedy rovná jazyku L, který má 6 slov.

Bodování:

Kdo neuvedl alespoň trochu slušný postup a napsal pouze výsledek, dostal celkem 0 bodů.
Kdo v co-L zapomněl na \epsilon, dostal celkem 0 bodů.
Kdo měl co-L dobře, ale (co-L)^2 výrazně špatně (např. slova délky 6 a větší), dostal celkem 1 bod.
Kdo měl dobře (co-L)^2, ale výsledek výrazně špatně (mám dojem, že takové řešení se vyskytlo jenom 1), dostal celkem 2 body.
Kdo v (co-L)^2 zapomněl na \epsilon a tím pádem výsledný jazyk měl o jedno slovo navíc, dostal celkem 4 body.
Plný počet 8 bodů dostal pouze ten, kdo napsal že výsledný jazyk se rovná jazyku L a uvedl alespoň náznak postupu, jak k tomu došel.

Podzim 2008, 20.11.2008, Vnitro
Příklad 03

Vašou úlohou bolo nájsť deterministický konečný automat, ktorý rozpoznáva jazyk všetkých slov nad abecedou {a,b,c}, ktoré obsahujú podslová aa a c (varianta 1) alebo podslovo ca alebo bb (varianta 2).

Automat nemusel byť vôbec minimálny, takže stačilo ak ste nakreslili akýkoľvek deterministický automat, ktorý rozpoznával uvedený jazyk. Okrem obrázka ste nemuseli špecifikovať nič ďalšie. Samozrejme riešenia zadané tabuľkou alebo podrobnou definíciou prechodovej funkcie boli akceptované tiež (akurát som si ich musel prekresliť, aby som to lepšie videl ;-))

Za väčšinu chýb (nesprávne alebo chýbajúce prechody, neoznačený počiatočný stav a akceptujúce stavy) som strhával bod za každú. Neplatilo to obecne vždy. Napríklad za chýbajúcu "slučku" na akceptujúcom stave som nestrhol 3 body ale 2 (aj keď technicky šlo o tri chýbajúce prechody). Občas, keď bola chyba veľmi závažná, tak ste mohli prísť aj o viac bodov.

Častou chybou bolo, že ste uviedli nedeterministický automat. V tom prípade ste mohli získať maximálne 6 bodov. Niektoré riešenia obsahovali nedeterminizmus napríklad iba na jednom mieste, v tom prípade som príslušný prechod pokladal za zlý a strhol iba 1 bod. Obecne bol veľký rozdiel medzi návrhom nedeterministického automatu a toho, kde sa nedeterminizmus dostal vďaka nepozornosti.

Špecificky ešte pre variantu jeden (podslová aa a c) ste dostali 5 bodov, keď ste uvažovali iba jedno z možných usporiadaní oboch podslov vo výslednom slove (teda, že automat akceptoval iba slová typu E.aa.E*.c.E* alebo iba typu E*.c.E*.aa.E*) a 4 body, keď ste namiesto oboch podslov vyžadovali iba jedno. Podobne, ak ste vo variante 2 vynucovali obe podslová.*

V prípade, že automat bol naozaj veľmi vzdialený správnemu riešeniu, tak som sa snažil nájsť aspoň nejakú myšlienku, ktorá by naznačovala, že ste navrhovali automat rozpoznávajúci jazyk zo zadania. Bohužiaľ, nie vždy to šlo.

Ešte jedna poznámka. Niektorí z vás používali alternatívne označenie akceptujúceho stavu (šípka smerujúca von). V tom by žiadny problém nebol. Problémom bolo, ak ste tak nerobili konzistentne a mali ste napríklad v jednom automate niektoré stavy označené tak, niektoré inak a niektoré pre istotou oboma spôsobmi.

Podzim 2008, 20.11.2008, Vnitro
Příklad 04

Varianta 1:
(a) L je regulárny a L = L prienik R => R je regulárny
Tvrdenie neplatí. Stačilo nájsť protipríklad.
Napríklad: L = \emptyset a R = {a^n.b^n|n>=0}.
(b) L je konečný a L.R nie je regulárny => R nie je regulárny
Tvrdenie platí. Dôkaz sporom:
- Predpokladáme platnosť negácie zadaného tvrdenia:
- L je konečný a L.R nie je regulárny a R je regulárny
- Keďže L je konečný, je aj regulárny, lebo všetky konečné jazyky sú regulárne.
- Keďže trieda regulárnych jazykov je uzavretá na zreťazenie a L a R sú regulárne, tak aj L.R musí byť regulárny jazyk. -> SPOR s predpokladom, že L.R nie je regulárny.

Varianta 2:
(a) L je konečný a L u (zjednotenie) R nie je regulárny => R nie je regulárny
Tvrdenie platí. Dôkaz sporom:
- Predpokladáme platnosť negácie zadaného tvrdenia:
- L je konečný a L u R nie je regulárny a R je regulárny
- Keďže L je konečný, je aj regulárny, lebo všetky konečné jazyky sú regulárne.
- Keďže trieda regulárnych jazykov je uzavretá na zjednotenie a L a R sú regulárne, tak aj LuR musí byť regulárny jazyk. -> SPOR s predpokladom, že LuR nie je regulárny.
(b) L je regulárny a R = L prienik R => R nie je regulárny
Tvrdenie neplatí. Stačilo nájsť protipríklad.
Napríklad: L = E* a R = {a}*.{b}* (alebo R = \emptyset)

Bodovanie 1(a) a 2(b):
2 body za správnu odpoveď, zvyšok za protipríklad. Ak bol protipríklad zlý alebo chýbal, tak ste ešte mohli získať 1 až dva body za "myšlienku", prečo to neplatí. Častou argumentáciou bolo, ak R je podmnožina regulárneho jazyka L, tak R musí byť regulárny. Samozrejme to neplatí, napríklad: L = E a R = {a^n.b^n|n>=0}. Opačnou argumentáciou bolo, ak L = L prienik R, tak aj R je regulárny. Opäť to neplatí, stačí si za L zobrať \emptyset a za R ľubovoľný neregulárny jazyk (ale určite nájdete aj mnoho iných protipríkladov). V takomto prípade ste dostali body iba za odpoveď. Rovnako som nedával žiadne body sa informáciu, že regulárne jazyky sú uzavreté na prienik, pretože ak to aj niekto vo svojej argumentácii použil, tak vždy nesprávne a prišiel k zlému záveru.*

Bodovanie 1(b) a 2(a):
2 body za správnu odpoveď, 1 za uvedenie, že konečné jazyky sú regulárne, 1 za informáciu o uzavretosti na zreťazenie (zjednotenie) a ten posledný, že ste to správne dali dokopy a došli k záveru. K tomu poslednému bodu. Možnože niektorí z vás budú mať pocit, že o ten bod prišli neprávom. Verte mi, strávil som veľa času premýšľaním, či vám dať 4 alebo 5 bodov. Väčšinou bol problém, že síce ste mali všetky fakty správne, ale nedošli ste k jasnému záveru (v tomto prípade jasnému sporu) a teda človek musel nejakú dobu premýšľať, kým dospel k tomu, prečo to vlastne plynie z toho, z čoho tvrdíte, že to plynie. Také riešenie som zväčša pokladal za nie úplne dokončené.
Ja k tomu záveru prísť viem, ale chcel som, aby ste ukázali, ako k nemu viete prísť vy, alebo ako to vy vidíte, prečo je to práve tak. Ale zase, veľmi záležalo od konkrétneho riešenia a od toho, ako konkrétne sa argumentovalo. Samozrejme, dôkaz sporom nebol jediným spôsobom ako to dokázať. Mnohí využili napríklad obmenu implikácie.

Po bode ste ešte mohli rôzne stratiť za nepravdivé tvrdenia, ktoré ste používali vo svojej argumentácii. Napríklad nesprávne znegovaná zadaná implikácia alebo jej obmena, tvrdenia typu reg.j. zjednotený (zreťazený) s nereg.j. implikuje, že výsledný jazyk nie je regulárny a podobne.

Veľmi ma prekvapilo, koľkí nevedia, že všetky konečné jazyky sú regulárne. Tiež koľko priestoru ste venovali, aby ste to zdôvodnili. Častokrát zložito (a nie vždy úplne správne) cez M-N vetu. Pritom fakt, že ak L a R sú regulárne, tak že aj LuR je regulárny, často nestál ani za zmienku, že to plynie z uzavrenosti regulárnych jazykov na zjednotenie.

Chcem vás poprosiť, aby ste v príklade, ako je tento, vždy jasne označili, či dané tvrdenie podľa vás platí alebo nie. Často bola argumentácia taká zmätočná, že sa to nedalo jasne zistiť. Občas sa dokonca stávalo, že argumentácia bola úplne správne, ale záver, ktorý z nej vyplynul, bol nesprávny.

Posledná poznámka sa týka všeobecne dôkazov a hlavne dôkazu sporom.
- Ak dokazujem správnosť niečoho, nestačí mi ukázať iba jeden príklad, kedy to platí.
- Nájdenie protipríkladu a tým zdôvodnenie nepravdivosti istého tvrdenia nie je dôkaz sporom.
- Dôkaz sporom sa používa na dokazovanie pravdivosti nejakého tvrdenia tak, že ukážeme neplatnosť jeho negácie pre všetky prípady. Z toho tiež plynie, že nestačí nájsť jeden protipríklad, kedy tá negácia neplatí. Tým dokážete iba to, že vaše pôvodné tvrdenie niekedy platí (určite aspoň pre ten konkrétny príklad), ale nie, že platí vždy. Overiť si to môžete napríklad na tvrdeniach 1(a) a 2(b), kde veľmi ľahko nájdete protipríklad na ich negáciu, ale pôvodné tvrdenia všeobecne neplatia.

Podzim 2008, 20.11.2008, Vnitro
Příklad 05

Vaším úkolem v tomto příkladu bylo nalézt k zadanému KA ekvivalentní automat v kanonickém tvaru. Celkem jste mohli získat až 12 bodů. Existovaly 2 varianty zadání, které se lišily pouze pojmenováním dvou stavů, výsledný automat v kanonickém tvaru tedy v obou případech vyšel stejně:

		a	b
<->	A	A	B
	B	C	A
<-	C	D	E
	D	D	D
	E	A	A

Nejlepším (a jak se ukázalo, také jediným úspěšným) řešením bylo použití standardních algoritmů známých z přednášky.

bodování:
Chybějící nebo chybně provedená kanonizace: -3 body.

Mnozí si neuvědomili, že vstupem algoritmu pro minimalizaci musí být automat s totální přechodovou funkcí. Za chybějící "ztotálnění": -7 bodů.

Chybějící vyznačení počátečního a koncových stavů u výsledného automatu: -5 bodů (pokud jste počáteční a koncové stavy měli vyznačeny u automatu, který byl výstupem algoritmu pro minimalizaci, odečítal jsem pouze 2 body).

Chybná "inicializace" algoritmu pro minimalizaci: -5 bodů. Nejčastěji se objevovala chyba, kdy jste na začátku ztotožnili stav 1 (který byl zároveň počáteční i koncový) s nekoncovými stavy, přičemž podle algoritmu musí být takový stav na začátku ztotožněn s ostatními koncovými stavy. Stejně tak bylo chybou, pokud jste na začátku vytvořili třídu obsahující pouze stav 1 (zadání bylo vymyšleno tak, aby i po minimalizaci zůstal stav 1 ztotožněn s nějakým jiným stavem). Naopak jsem toleroval, pokud jste na začátku vytvořili samostatnou třídu obsahující pouze nově přidaný stav ("černou díru"). V tomto případě jste se i tak mohli dopracovat ke správnému výsledku, neboť hned v prvním kroku algoritmu by se tento nový stav "osamostatnil". Obecně to však takto nefunguje (zkuste se zamyslet nad protipříkladem) a nový stav musí být na začátku ztotožněn s ostatními nekoncovými stavy.

Drobné formální nedostatky, zmatený zápis, chyby z nepozornosti: -0 až -3 body dle (subjektivně hodnocené) závažnosti.

Vážnější chyby v algoritmu pro minimalizaci: -3 body za každou chybu (např. rozdělení dvouprvkové třídy do dvou jednoprvkových, přestože podle předchozí tabulky měla být původní třída zachována atp.).

Někteří jste se pokoušeli příklad řešit jiným způsobem, bohužel se mi z Vašeho zápisu nepodařilo zjistit jakým. (Mám podezření, že jste se pokoušeli napočítat prefixovou ekvivalenci pro jazyk rozpoznávaný zadaným automatem a s její znalostí pak vyrobit minimální automat. Takto samozřejmě lze také postupovat, ke správnému řešení se však nepřiblížil nikdo.)

Někteří se dopustili více chyb z výše uvedených. Nepostupoval jsem striktně tak, že bych za každý typ chyby odečetl příslušný počet bodů, vždy jsem se snažil dávat alespoň minimální počet bodů, viděl-li jsem náznak smysluplného postupu.

Myslím, že celkově dopadl příklad dobře, přesto mi odpusťte rýpavou poznámku: u mnoha řešení jsem nabyl dojmu (ze způsobu zápisu, či různých ne úplně smysluplných komentářů), že jste se sice naučili postup, který jste viděli na přednášce/cvičení (což je jistě dobře), avšak nechápete myšlenku, která se za tímto postupem skrývá, tedy "co se vlastně při provádění algoritmu děje". Mezi cíli tohoto předmětu určitě není biflování spousty různých algoritmů, ale právě pochopení jejich principu. Vemte prosím tento fakt na vědomí při přípravě na závěrečnou písemku :).

Podzim 2008, 20.11.2008, Vnitro
Příklad 06

Vazeni studenti Vasim ukolem bylo rozhodnout a dokazat zda je jazyk
L = {a^ib^j^c^k | i>=2, 0<=j<=k} je regularni (v alternativnim prikladu to byl jazyk
L = {a^ib^j^c^k | i>=2, 0<=j<k} .

Za tento priklad se dalo ziskat 15 bodu.
Pouziti M.N. resp. N. vety:
Predpokladame, ze jazyk L je regularni a tudiz dle M.N. vety existuje n \in N
takove ze |~L|=n. Uvazme n+1 slov:
aa,aab,aab^2,...,aab^n
Z predpokladu ze |~L| = n musi existovat aspon jedna dvojice slov aab^i a aab^j
takova ze aab^i ~ aab^j (jsou v relaci). Bez ujmy na obecnosti muzeme
predpokladat, ze i<j.
Z definice prefixove ekvivalence (~L) plyne ze
pokud aab^i ~ aab^j pak aab^ic^i \in L prave tehdy kdyz aab^jc^i \in L.
Jelikoz slovo aab^ic^i patri do L a aab^jc^i nepatri do L (jelikoz j>i), pak
slova aab^i a aab^j nejsou v relaci a tudiz |~L| > n coz je spor s predpokladem
ze (|~L| = n).
Timto jsem ukazali ze **|~L|** je nekonecno a tudiz jazyk L neni regularni.

Pozn. Pro druhou variantu prikladu by se priretezovalo slovo c^(i+1).

Caste chyby:
Nepresna argumentace ze |~L| = nekonecno - srazka 1 az 5 bodu
Pri pouziti N. vety misto M.N. vety nedostatecna argumentace, proc dochazi ke sporu - srazka 1 bod
Spatne pouziti M.N resp N. vety (typicke chyby: uvazovali jste slova, ktera byla v relaci ci jste nedokazali, ze |~L| = nekonecno) - ziska max 7 bodu

Pouziti P.L.
Necht n\in N je libovolne nadale pevne.
Zvolime slovo w = a^2b^nc^(n+1).
Uvazme vsechna rozdeleni slova w na podslova x,y,z splnujici podminky P.L.
(|xy|<=n a |y|>0)
Pro vsechna rozdeleni, kde se v podslove y nachazi aspon jedno pismeno 'a'
zvolime pumpovaci konstantu i=0.
Pak slovo xy^iz nepatri do L jelikoz #a(w) < 2.
Pro vsechna rozdeleni, kde se v podslove y nachazi aspon jedno pismeno 'b' a
neobsahuje 'a' zvolime pumpovaci konstantu i=3.
Pak slovo xy^iz nepatri do L jelikoz #b(w) > #c(w).
Jelikoz jsme ukazali ze pro vsechna platna rozdeleni (splnuji podminky P.L.)
existuje i \in N_0 takove, ze xy^iz nepatri do L, pak z P.L. plyne, ze jazyk
neni regularni.

Pozn. Tento dukaz projde pro obe variaty zadaneho jazyka L.

Caste chyby:
Pouze odpoved, ze jazyk neni regularni bez dukazu - zisk 2 bodu

Spatne pouziti ci nepochopeni P.L. Zde byla siroka skala chyb: vubec jste nezvolili slovo w a uvedli jste jen nejak rozdeleni, volili jste nekolik slov nebo jste je parametrizovali nejakymi promennymi ruznymi od n a podobne. Typicky jste mohli ziskat 2 az 6 bodu.

K dukazu jste pouzili predpoklad, ze kdyz L1 je regularni a L2 neni regularni pak L1.L2 neni regularni. (toto samozrejme neplati) - za toto reseni jste mohli zikat max 6 bodu (pokud P.L. pro L2 bylo zcela spravne)

Spatne zvolene slovo (typicky w = a^nb^nc^n a jina). Pro tyto slova dukaz nesel udelat. Vase hodnoceni pak zalezelo na tom, jaka jste volili rozdeleni a jak jste argumentovali, ze jazyk neni regularni. Typicky jste pouzili neplatna rozdeleni (porusujici podminku |xy|<=n), kde podslovo y obsahovalo pismena 'b'. - zisk 4 az 7 bodu

Dobre zvolene slovo ale chybela nejaka rozdeleni:
pokud jste uvedli jen jedno konkretni rozdeleni - zisk 5 bodu
pokud jste uvedli jen rozdeleni, kde v podslove y byla pismena 'b' - zisk 8 bodu
pokud jste uvedli rozdeleni, kde v podslove y byla pismena 'b' a rozdeleni, kde v podslove y byla jen pismena 'a' (tudiz jste pumpovali i = 0 a i > 1) - zisk 10 bodu
pokud vam chybely treba jen rozdeleni, kde podslovo x = 'a' a vse ostatni jste tam meli - zisk 12 bodu
pokud vam jen chybelo rozdeleni x=epsilon, y = a, z = zbytek - zisk 14 bodu

Dale se objevovaly chyby, ktere ne vzdy (zalezelo na kontextu celeho Vaseho reseni) byly penalizovany. Typicky jste uvazovali i neplatna rozdeleni (porusujici podminky P.L.) nebo jste volili slovo w= aab^(n-2)c^n, ktere pro n = 1 nema vyznam, neuvedli jste pumpovaci konstantu n, vubec jste nenapsali co delate a podobne.

Tento vycet chyb neni uplny a uvedene zisky bodu jsou jen priblizne. Vase reseni obcas obsahovala ruzne kombinace uvedenych chyb a podle toho bylo i hodnoceno.

Na zaver bych rekl, ze jsem byl zklaman jak tento priklad dopadl a rad bych od Vas slysel cim to bylo. Jestli tim ze ten priklad byl pro Vas tak tezky nebo jste nepochopili P.L. resp M.N. vetu nebo jste jen podcenili pripravu na pisemku.

Podzim 2008, 20.11.2008, Vnitro
Příklad 07

c): definujte pravou kongruenci ~ takovou, ze sjednocenim nekterych trid rozkladu Sigma/~ je L = {a}.({a,b}.{a,b})*. ~ je ruzna od ~L*

** urcete index ~L: spravny vysledek 4 + zduvodneni, jak jste k nemu dosli, tj. nejlepe nakresleny TOTALNI automat akceptujici jazyk L + zduvodneni, ze je to MINIMALNI automat. Chybejici zduvodneni jsem nakonec odpoustela, pokud bylo videt, ze tomu rozumite, ale odpoved 4 skutecne nestacila. Ti, kdo napsali, ze index je nekonecno, meli rovnou nulu a meli by se stydet, to byl prece jasne regularni jazyk a ten musi mit konecny index. Casta odpoved byla 3, protoze automat nebyl totalni - za to byly tusim 3 body. Podobne byly bodovany vysledky, ke kterym jste dosli na zaklade automatu, ktere akceptovaly L, ale nebyly minimalni.*

** popsat tridy rozkladu podle ~L: odpoved 'viz automat' nestacila, podobne definovani trid pomoci jednoho reprezentanta neni postacujici - potrebovala jsem videt, ze skutecne vite, jaka slova jsou v ktere tride. Spravny popis byl napr. regularni vyraz pro kazdou tridu nebo slovni popis. Pokud byly tridy popsany zcela spatne, nedavala jsem ani 1 bod za spravne urceni poctu trid. Jinak byl pocet bodu umerny poctu spravne popsanych trid, take jsem zohlednovala, jestli popis aspon odpovida nakreslenemu automatu (pokud ano, bylo videt, ze tomu rozumite, jen jste napr. zapomneli ztotalnit automat).*

** definice prave kongruence: nejjednodussi spravna odpoved byla identita, coz se v jednom reseni objevilo. Jinak spravnych reseni existuje rada, velka cast odpovedi vsak ani nepopisovala relaci na Sigma*, ale nejakou mnozinu slov, napr. sjednoceni trid rozkladu podle ~L. Ti, kteri psali relaci, pak casto udelali chybu zminovanou v predchozim prispevku, tj. uvazovali v definici relace jen slova, ktera patri do jazyka. Prava kongruence vsak musi byt relace ekvivalence, tj. musi byt napr. reflexivni - vsechna slova jazyka musi nejak rozdelit do trid.*

Celkove jsem take byla kvalitou reseni dost zklamana, zda se mi, ze priklad nebyl tezky, stacilo rozumet definici ~L a prave kongruence...

Podzim 2009, 20.12.2009, Vnitro

Definujte *nedeterministický konečný automat s ε-kroky* a funkci $D_\varepsilon : Q \longrightarrow 2^Q$.

Příklad 1
8 bodů

NFA $\mathcal{M}$ je pětice $\mathcal{M}(Q, \Sigma, \delta, I, F)$ kde

Správne riešenie:
NFA s ϵ-krokmi je pätica $\mathcal{M} = (Q, \Sigma, \delta, q_o, F)$, kde
Q- je neprázdna konečná množina stavov
Σ- je konečná množina vstupných symbolov alebo vstupná abeceda
δ- je prechodová funkcia definovaná ako totálne zobrazenie $\delta : Q \times (\Sigma \cup \{\epsilon\}) \to 2^Q$
$q_0 \in Q$- je počiatočný stav
$F \subseteq Q$- je množina koncových stavov

Definícia $D_\epsilon : Q \to 2^Q$:
Pre dané $p \in Q$ je $D_\epsilon(p)$ najmenšia množina $X \subseteq Q$ taká, že platí:

- $p \in X$
- Ak $q \in X$ a $r \in \delta(q, \epsilon)$, potom aj $r \in X$

Tých 8 bodov bolo rozdelených tak, že 6 pripadlo na definíciu automatu a 2 na funkciu D_ϵ.
Pri automate ste dostali bod za "pätica" a potom po bode za význam každej jej položky. Uznával som aj alternativnu definícu, kedy sa namiesto jedného počiatočného stavu q_0 uvažuje množina $I \subseteq Q$ počiatočných stavov. Ak ste neuviedli, že ide o konečnú množinu stavov a konečnú abecedu, tak ste prišli o bod. Rovnako o bod ste prišli, ak ste usporiadanú päticu zapísali v množinových zátvorkách alebo do abecedy Σ vkladali prázdne slovo ϵ.
Častou chybou, ktorú som ale nepenalizoval bolo, že ste o δ tvrdili, že ide o parciálnu prechodovú funkciu. Pri NFA a rovnako aj NFA s ϵ-krokmi ide o totálne zobrazenie.
Funkciu D_ϵ ste sa často snažili definovať slovne. Bohužiaľ z toho nebolo zvyčajne zjavné, ako sa počíta, respektive či p patrí do $D_\epsilon(p)$ alebo nie, preto to bolo ohodnotené iba jedným bodom. Častou chybou bola snaha definovať ju pomocou rozšírenej prechodovej funkcie $\hat{\delta}$. Problém je, že $\hat{\delta}$ sa definuje pomocou D_ϵ. Rovnako sa objavilo veľa iných rôznych definícií s použitím prechodovej funkcie δ, ktoré bohužiaľ definovali iba množinu stavov, do ktorých sa z daného stavu p dostanem po jednom/dvoch ϵ-krokoch a ešte niekoľko "cyklických" definícií, kde sa $D_\epsilon(p)$ počítalo pomocou $D_\epsilon(q)$ nejakého iného stavu q, kde ale dochádzalo k tomu, že pre isté automaty by sa výpočet D_ϵ zacyklil.
Posledná a asi najčastejšia chyba s D_ϵ, na ktorú chcem upozorniť bolo, že ste ju často zamieňali za prechodovú funkciu.

Podzim 2009, 20.12.2009, Vnitro

Určete, kolik slov má jazyk $L^3 \cap L^2$, kde $L = \{cd, cc, c, d\}$. Všechna slova vypište. **Příklad 2**

5 bodů

$L^3 = L . L^2 =$ $\Rightarrow L^3 \cap L^2 = L^2$ neplatí

Určete, kolik slov má jazyk $L^3 \cap L^2$, kde $L = \{cd, cc, c, d\}$. Všetky slová vypíšte.

Správne riešenie:

$L^2 = \{cdcd, cdcc, cdc, cdd, cccd, cccc, ccc, ccd, cc, cd, dcd, cdd, dc, dd\}$

Teraz si stačilo uvedomiť, že v L^3nebudú nikdy slová kratšie ako 3 a zároveň sa zamyslieť, ktoré zo zvyšných slov v L^2budú aj v L^3(teda môžu vzniknúť zreťazením niektorých 3 slov z jazyka L). Z toho ste dostali 10 slov.

$L^3 \cap L^2 = \{cdcd, cdcc, cdc, cdd, cccd, cccc, ccc, ccd, dcd, dcc\}$

Za každú úplnu dvojicu správnych slov ste dostali bod. Respektíve za každú začatú dvojicu chybných slov ste o jeden prišli. Veľmi častou chybou bolo použitie nepravdivého tvrdenia $L^3 \cap L^2 = L^2$. Vtedy ste získali maximálne 2 body, ale aj to len vtedy, ak ste mali správne jazyk L^2. Často ste ešte zabúdali, že v množine nemôžu byť duplicitné prvky, vtedy som strhával bod. Obľúbenou chybou tiež bolo zahŕňanie slov c, d *a* ϵ *do* L^2.

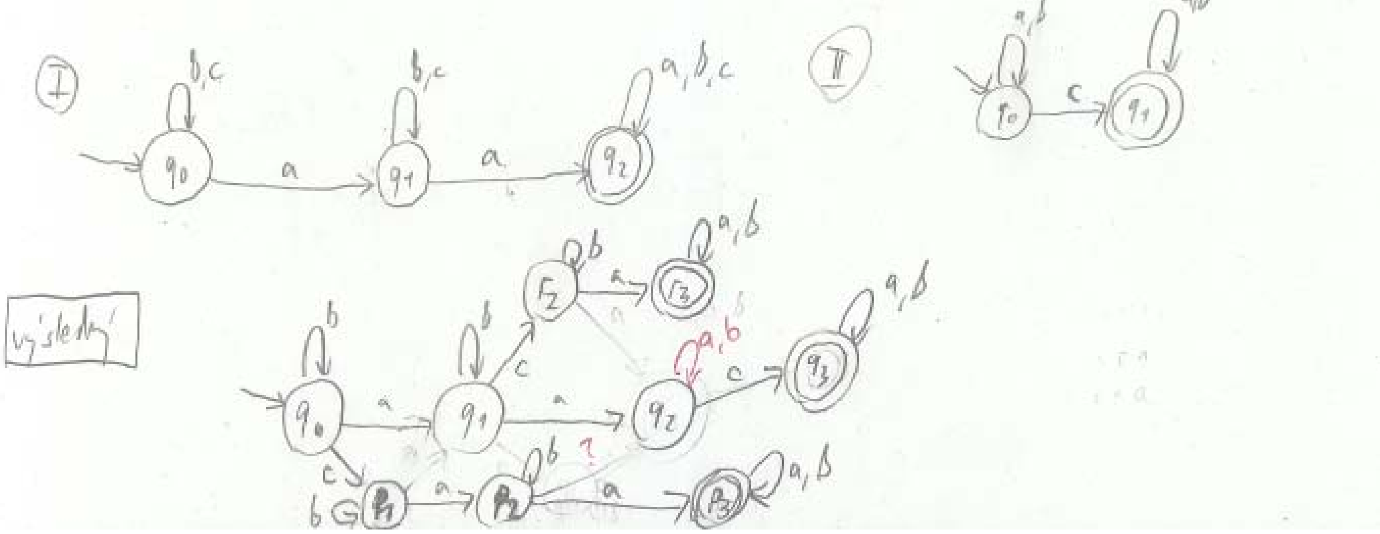
Podzim 2009, 20.12.2009, Vnitro

Navrhněte *deterministický* automat generující jazyk

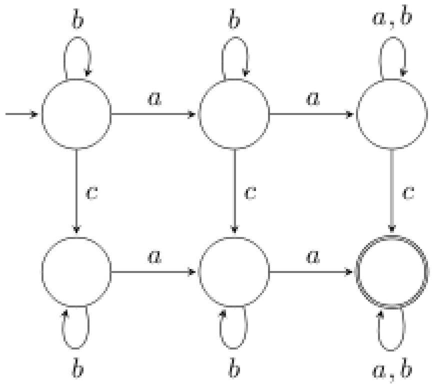
Příklad 3
10 bodů

5

$$L = \{w \in \{a,b,c\}^* \mid \underbrace{\#_a(w) \geq 2}_{I},\ \underbrace{\#_c(w) = 1}_{II}\}.$$



Řešení: mohlo vypadat např. takto:



Hodnocení: *Za správný automat 10 bodů. Za drobné nedostatky jsem strhával většinou 2 body (chybějící počáteční/koncový stav), za větší nedostatky 4 a více bodů. Byl-li automat úplně nesmyslný, ale aspoň* deterministický, *hodnotil jsem jedním bodem za snahu. Naopak,* nedeterministické automaty *(automat, který obsahuje přechod pod epsilon,* je *nedeterministický) mohly dostat nejvýše dva body, byly-li jinak dobře.*

Podzim 2009, 20.12.2009, Vnitro

Rozhodněte, zda platí následující implikace. Svá rozhodnutí zdůvodněte.

Příklad 4
10 bodů

(a) $L.L$ není regulární $\Longrightarrow$ L není regulární

(b) L je regulární $\Longrightarrow$ $\{w.w \mid w \in L\}$ je regulární

Řešení: a) platí**, b)** neplatí. Zdůvodnění:

a) Kdyby Lbylo regulární, pak by i $L \cdot L$muselo být regulární protože třída regulárních jazyků je uzavřená na operaci zřetězení. Jestliže tedy $L \cdot L$regulární není, pak není regulární ani L.

b) Vezměme si například jazyk $L = \{a\}^* \cdot \{b\} = \{a^n b \mid n \in \mathbb{N}_0\}$. Ten je zřejmě regulární (dá se zapsat regulárním výrazem). Jazyk $\{w \cdot w \mid w \in L\} = \{a^n b a^n b \mid n \in \mathbb{N}_0\}$ovšem regulární není (dá se snadno dokázat buďto pumping lemmatem nebo Myhill-Nerodovou větou, ale to jsem po vás nechtěl). Implikace b) proto neplatí.

Hodnocení: *Za odpověď platí/neplatí 1 bod, za zdůvodnění 4 body, dohromady 10 bodů. Odpověď na a) naprostá většina zvládla správně a typicky bylo zdůvodnění buď dobře nebo úplně špatně, u několika písemek jsem strhl jeden nebo dva body (ve většině případů to bylo tak, že kromě správného zdůvodnění bylo na listu ještě nějaké nepravdivé tvrzení, za to jsem ten bod strhl). Odpověď na b) naprostá většina nezvládla, ti, co ji měli, měli buď žádné nebo úplně špatné zdůvodnění (0b), úplně správné zdůvodnění (4b) nebo náznak správného protipříkladu, ale formálně špatně (2-3b, většinou 2b).*

Seznam (až příliš) častých nepravdivých tvrzení a jiných chyb:

- $\{w \cdot w \mid w \in L\}$*je totéž, co* L^2.
- *Neregulární jazyky jsou uzavřené na zřetězení.*
- *Je-li* L^2*regulární, je i* L*regulární.*
- *Je-li slovo w z regulárního jazyka, pak je taky regulární. (Co by to mělo být regulární slovo?)*
- *Každý regulární jazyk je konečný.*
- *Jazyk je nekonečný, a proto obsahuje nekonečné slovo.*
- *Negací implikace* $A \Rightarrow B$*je implikace* $A \Rightarrow \neg B$.

(Bylo by skvělé, kdybyste se na tento seznam podívali a uvědomili si, proč žádná z těchto vět neplatí.)

Podzim 2009, 20.12.2009, Vnitro

Rozhodněte, zda je jazyk $L = \{a.b^n.c^n \mid n \text{ je liché}\}$ regulární. Své tvrzení dokažte. *(Pro důkaz, že jazyk je regulární, stačí napsat odpovídající gramatiku nebo automat.)*

Příklad 5
15 bodů

Mozne reseni pomoci P.L.
Necht n je libovolne.
Zvolim slovo w = ab^{2n+1}c^{2n+1}.
Nyni uvazim vsechny rozdeleni splnujici P.L.
a)
x = epsilon
y = ab^k
z = b^{2n+1-k}c^{2n+1}
Kde k>=0 a k<n.
Pak pro i = 0 dostavame, ze xy^iz = b^{2n+1-k}c^{2n+1} nelezi v L, jelikoz neobsahuje 'a'.
b)
x = ab^k
y = b^l
z = b^{2n+1-k-l}c^{2n+1}
Kde k>=0, l>0 a l+k<n.
Pak pro i = 0 dostavame, ze xy^iz = ab^{2n+1-l}c^{2n+1} nelezi v L, jelikoz l>0 a tudiz pocet 'b' je ostre mensi nez pocet 'c'.
Z P.L. dostavame ze L neni regularni.
- -
Mozne reseni pomoci M.N. vety.
Predpokladejme, ze L je regularni a tudiz dle M.N vety je index ~L konecny a tak existuje n \in N, takove ze |~L| = n.
Uvazime n+1 slov:
ab, ab^3, ab^5, ... ,ab^{2n+1}
Nyni ukazeme, ze zadne dve slova nejsou ve ~L (nejsou v stejne tride rozkladu podle ~L)
Necht 1<=i,j<=2n+1 a i,j jsou liche.
Pak plati:
ab^ic^i lezi v L
ab^jc^i nelezi v L
Tudiz dle definice prefixove ekvivalence (~L) plati:
ab^i neni v relaci s ab^j. Dostavame tedy, ze |~L| > n, coz je spor s nasim predpokladem (|~L| = n) a tudiz L neni regularni.
- -
Hodnoceni:
spravne reseni pomoci P.L. nebo M.N. vety - 15 bodu
drobne chyby v P.L. nebo v M.N vete - 13-15 bodu
(typicky jsem strhaval bod, kdyz bylo nekde > misto >= a tudiz jste neuvazili nejake konkretni rozdeleni; nebo jste si popletli N. vetu a M.N. vetu a podobne)
zavaznejsi chyby v P.L. nebo v M.N vete - 10-12 bodu (typicky za chybejici mnozinu rozdeleni, napr. kde x = epsilon; nepresne zduvodneni, ze |~L| = nekonecno, napr chybejici argument, ze ab^i neni v relaci s ab^j)
spatne zvolene slovo ab^nc^n - 8 bodu
spatne zvolene slovo ab^nc^n + drobna chyba - 7 bodu
spatne zvolene slovo ab^nc^n + chybejici mnozina rozdeleni - 4-5 bodu
chybejici volba slova - 3-4 body
spravna odpoved bez dukazu - 2 body
zadna ci nejasna odpoved nebo nesmyslna tvrzeni (napr jazyk neni konecny tudiz neni regularni) - 0-1 bod

Uvedene hodnoceni je jen orientacni a mnoha Vase reseni obsahovaly ruzne kombinace chyb, ktere byly podle jejich zavaznosti penalizovany.

Dalsi caste chyby byly napr:
- spatne pouziti uzaverovych vlastnosti
- volba slova, ve kterem jste uvadeli nespecifikovane promenne
- pouziti nedokazanych tvrzeni typu priretezenim 'a' nezmenime regularitu.

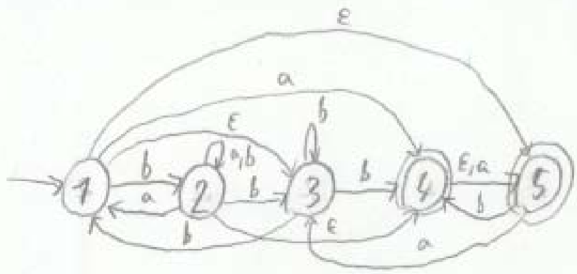
Takove chyby byly hodnoceny individualne dle jejich zavaznosti a s prihlednutim na zbytek reseni.

Podzim 2009, 20.12.2009, Vnitro

K zadanému konečnému automatu $\mathcal{A}$ sestrojte ekvivalentní (nedeterministický) konečný automat bez ε-kroků.

Příklad 6
12 bodů

$\mathcal{A}$	a	b	ε
$\to$ 1	{4}	{2}	{3,5}
2	{1,2}	{2,3}	{4}
3	$\emptyset$	{1,3,4}	$\emptyset$
$\leftarrow$ 4	{5}	$\emptyset$	{5}
$\leftarrow$ 5	{3}	{4}	$\emptyset$



odstranění ε kroků ⇒ podle grafu automatu ↗

	a	b
$\leftrightarrow$ 1	{3,4,5}	{1,2,3,4,5}
$\cancel{\leftarrow}$ 2	{1,2,3,4,5}	{2,3,4,5}
$\cancel{\leftarrow}$ 3	$\emptyset$	{1,3,4,5}
$\leftarrow$ 4	{3,5}	{4,5}
$\leftarrow$ 5	{3}	{4,5}

10b

List 4 je již opraven a za chvíli byste měli mít body v pozn. blocích. Příklad byl celkem za 12 bodů. Plný počet bodů obdržel ten, kdo uvedl nedeterministický automat ekvivalentní k zadanému automatu a uvedl

množiny . Za většinu chyb jsem strhával 2 body. Za spočítání správných množin a za označení stavu 1 za akceptující jste mohli získat po +2 bodech.

Časté chyby:

neoznačení stavu 1 za akceptující

špatně vypočítaná hodnota , konkrétně jste zapomínali na stav 5

špatně spočítaná př. funkce, většinou jste opomněli zahrnout nějaké to

Podzim 2009, 20.12.2009, Vnitro

Uvažujte následující relace na slovech nad abecedou $\Sigma = \{a, b\}$. U každé relace určete, zda se jedná o pravou kongruenci. Pokud rozhodnete, že se o pravou kongruenci nejedná, dokažte to. V opačném případě určete index relace a popište jednotlivé třídy ekvivalence.

Příklad 7
15 bodů

9

(a) $u \sim v \overset{def}{\iff}$ u obsahuje podslovo a právě když v obsahuje podslovo a

(b) $u \sim v \overset{def}{\iff}$ u obsahuje podslovo aa právě když v obsahuje podslovo aa

(c) $u \sim v \overset{def}{\iff} \#_b(u) \leq \#_b(v)$

Celkově bylo možno získat 15 bodů, z toho za část a) 6 bodů, b) 5 bodů a c) 4 body. Nerovnoměrné rozdělení bodů odpovídá nerovnoměrné náročnosti, v části a) bylo třeba zjistit, že je to kongruence a určit třídy, u b) stačilo zjistit, že to není kongruence, a konečně c) nebyla ani ekvivalence.
U všech odrážek bylo možno získat bod za prosté rozhodnutí je/není kongruence, další body byly za určení indexu s popsání tříd, resp. zdůvodnění, že se nejedná o kongruenci. U části a) nebylo v zadání vyžadováno, abyste zdůvodnili, proč je to kongruence, k dosažení plného počtu bodů to tedy nebylo třeba. Pokud jste ale zdůvodnění uvedli, mohli jste získat 1-3 body (podle kvality zdůvodnění) i v případě, že chyběly třídy rozkladu.

Nejčastější problémy:
** Rozhodování, zda relace je pravá kongruence. Většinou jste si pamatovali, že má platit podmínka u~v => u.w~v.w. Bohužel to mnoho z Vás dokazovalo asi takto: "u~v, pak u i v obsahují a (aa), ať přidáme cokoli, pořád budou v relaci". Chyba je samozřejmě v tom, že slova u,v mohou být v relaci také tehdy, když neobsahují a (aa). V případě a) je správné zdůvodnění toto: "u~v => u i v osahují 'a' NEBO u i v neobsahují 'a'. Pokud obě slova obsahovala 'a', přiřetězení slova w to nezmění a u.w bude opět v relaci s v.w. Pokud slova neobsahovala 'a', pak buď slovo w obsahuje 'a', pak i u.w a v.w obsahují 'a' a tedy jsou v relaci, nebo w neobsahuje 'a', pak u.w ani v.w neobsahují 'a' a proto jsou opět spolu v relaci."*
V části b) podobná úvaha neplatí, je totiž možné najít dvě slova, která nejsou v relaci, ale po přiřetězení stejného slova v relaci budou. Např. u=a, v=b, w=a.
** V části a) jste míchali dohromady popis tříd a relací. Relace je množina uspořádaných dvojic slov, zatímco třída rozkladu je množina slov, které jsou spolu vzájemně (každé s každým) v relaci. Takže v popisu tříd nedává smysl mluvit o tom, že "u obsahuje 'a' a zároveň v obsahuje 'a'", v horším případě "u obsahuje 'a' a v neobsahuje 'a'".*
** Nevšimli jste si, že c) není symetrická relace, případně jste se ani nezamýšleli, zda se jedná o ekvivalenci. Několik dobrodruhů by si mělo ujasnit, co znamená, že relace je reflexivní/symetrická/tranzitivní. Také je nesmysl tvrdit, že tato relace má nekonečně mnoho tříd rozkladu. O rozkladu množiny můžeme uvažovat pouze na základě relace ekvivalence (ta určuje, jak vypadají třídy), pokud ji nemáme, nemáme ani třídy.*

Podzim 2010 , 18.12.2010, Vnitro

Definujte *deterministický konečný automat.* Dále definujte, kdy jsou stavy tohoto automatu *jazykově ekvivalentní.* **Příklad 1** **8 bodů**

DFA je uspořádaná pětice (1 bod) (Q,Σ,\delta,q_0,F), kde
- Q je konečná neprázdná množina stavů (1 bod; pokud chyběla konečnost tak 0 bodů, pokud neprázdnost tak 0.5 bodu, pakliže jste později výslovně neuvedli, že počáteční stav je z Q),
- Σ je konečná vstupní abeceda (0.5 bodu; pokud jste uvedli pouze “abeceda”, tak jsem to taky uznal, protože dle definice z první přednášky je abeceda konečná množina),
- \delta: Q x Σ -> Q (1 bod) je parciální přechodová funkce (0.5 bodu),
- q_0 ∈ Q (0.5 bodu) je počáteční stav (0.5 bodu),
- F ⊆ Q (0.5 bodu) je množina koncových stavů (0.5 bodu).

Správná definice jazykové ekvivalence viz například slidy ke 4. přednášce či skripta, str. 27. Pokud byla z Vašeho zápisu patrná správná myšlenka ale pokulhávalo formální provedení, dával jsem za tuto část nejvýše 1 bod.

Časté chyby: Často jste zapomínali na konečnost množiny stavů, u _konečného_ automatu je to poměrně zásadní vlastnost. Vzhledem k tomu, že se jednalo o jednu z nejzákladnějších definic, strhával jsem (půl)body i za formální chyby typu množinových závorek místo kulatých či míchání znaků ⊆ a ∈. Dále bych chtěl podotknout, že množina koncových stavů může být prázdná, minimální automat pro prázdný jazyk je jednostavový bez koncových stavů. Stejně tak může být teoreticky prázdná i vstupní abeceda (neprázdnost abecedy v definici jsem však nepovažoval za chybu). Automat také nemá "neterminály" ani "pravidla". Za různé "okecávací" definice jsem dal maximálně 3 body podle toho, jak moc se taková definice blížila definici formální.

(Mimochodem, ony jsou nějaké studijní materiály, kde se o koncových stavech mluví jako o "konečných" stavech? Spousta z Vás to používá, ale zní to dost divně.)

U druhé části příkladu byl hlavní problém v tom, že jste příslušnou definici neznali. Někteří si popletli jazykovou ekvivalenci stavů s ekvivalencí automatů. Jiní si matně uvědomovali, že jazyková ekvivalence stavů nějak souvisí s relací prefixové ekvivalence pro jazyk akceptovaný daným automatem a pokoušeli se o intuitivní přístup přes "stejný potenciál k dokončení" (to se však týká slov, ne stavů; navíc je ta intuice příliš hrubá). Občas jste užívali pojmu "slova generovaná stavem", ten však nebyl nikde definován. (Pokud jste tím mysleli jazyk, který by automat akceptoval po prohlášení daného stavu za počáteční, pak jste byli na správné cestě, je ale třeba napsat to přesněji). Tyto pokusy byly za 0 bodů. Stejně tak jsem hodnotil 0 body definici, ve které byla podmínka příliš silná, například že "z obou stavů musíme pod stejným znakem/slovem přejít do stejného stavu." Stejně tak není pravda, že stavy jsou jazykově ekvivalentní právě když se do nich "dovedeme z q_0 dostat pod stejným slovem" - u DFA by to implikovalo, že jsou tyto stavy stejné. Jako formální chyby, za které jsem strhával nejvýše 1 bod (pokud byl zbytek dobře), bych uvedl například chybějící univerzální kvantifikátor či záměnu spojek ∧ a <=>.

Podzim 2010 , 18.12.2010, Vnitro

Určete, kolik slov má jazyk $(co{-}L) \smallsetminus L^2$, kde $L = \{a, a^4\} \cup \{a^7\}.\{a\}^*$ je jazyk nad abecedou $\{a\}$. Všechna slova vypište. **Příklad 2**
5 bodů

Vaším úkolem bylo vypsat slova jazyka (co-L)\L^2, kde L={a,a^4}∪{a^7}{a}* je jazyk nad abecedou {a}. Tento jazyk obsahuje přesně tři slova a to \epsilon,a^3 a a^6. Stačí si uvědomit, že co-L={\epsilon,a^2,a^3,a^5,a^6}, přičemž posléze rozebráním jednotlivých případů zjistíme, že slova \epsilon, a^2 a a^5 nelze získat jako zřetězení dvou slov z jazyka L, zatímco slova a^2=a.a a a^5=a.a^4 ano. Za každé chybějící či přebývající slovo jsem strhával 1.5 bodu.

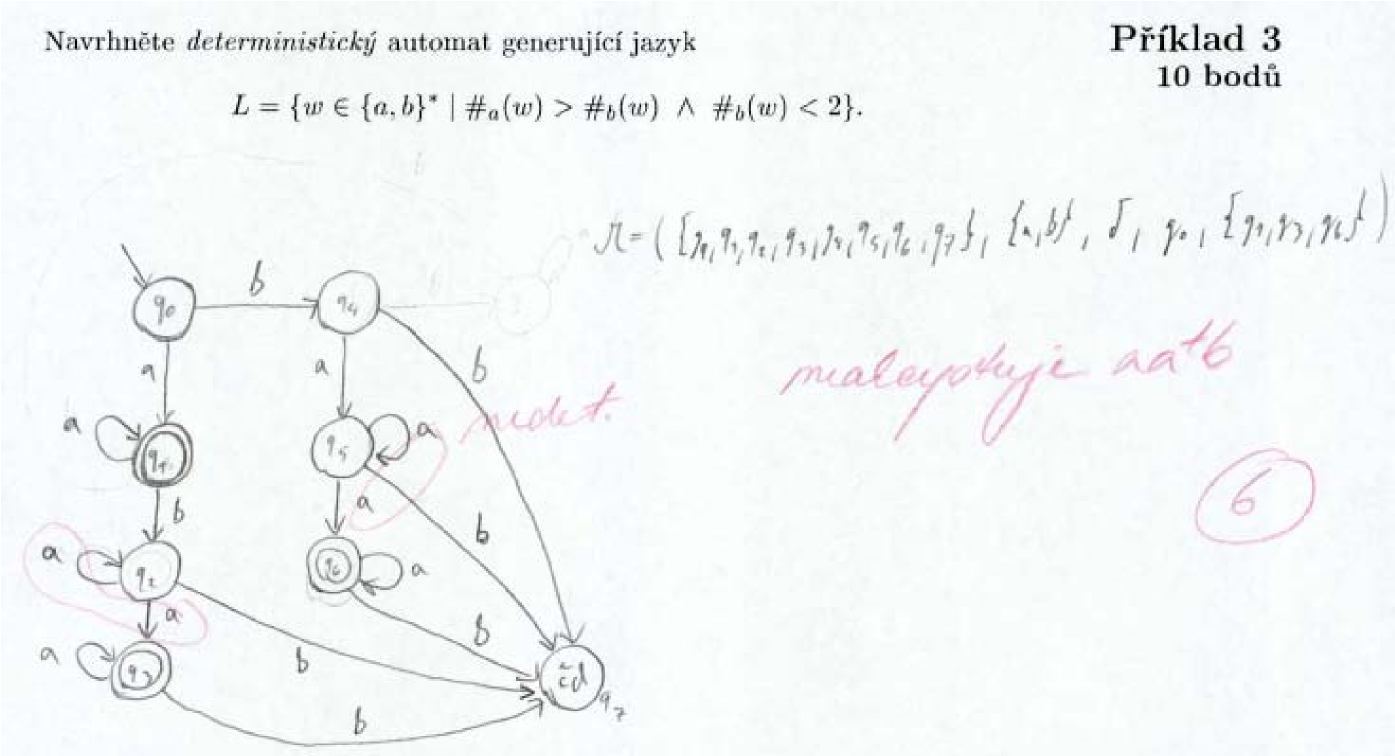
Časté chyby: Drtivá většina měla příklad správně. Občas jste nepostřehli, že co-L obsahuje \epsilon. (Někteří bůhvíproč pochybovali, zda je prázdné slovo taky slovo. Je.) Někteří si nevšimli, že L obsahuje alovo a^7 = a^7.\eps ∈ {a^7}{a}. Pokud jste napsali, že L=Σ* a tedy co-L=∅, žádné body jste bohužel nedostali.*
(EDIT: Jo a neškrtejte prosím ve výsledcích, které se dají rychle napsat znovu a bez škrtání, pak se v tom nedá vyznat.)

Po sečtení bodů z celého listu jsem případné půlbody zaokrouhlil nahoru.

Celkově to, myslím, nedopadlo špatně, nicméně vzhledem k tomu, že první příklad se týkal zcela zásadní definice, mohlo to dopadnout i lépe. Věnujte proto před závěrečnou písemkou dost času učení (a pochopení) definic a vět z přednášky, ať se zbytečně nepřipravujete o "body zadarmo."

Případné dotazy či reklamace můžete psát do tohoto vlákna.

Podzim 2010 , 18.12.2010, Vnitro



Navrhněte *deterministický* automat generující jazyk

Příklad 3
10 bodů

$$L = \{w \in \{a, b\}^* \mid \#_a(w) > \#_b(w) \ \wedge\ \#_b(w) < 2\}.$$

Asi polovina z Vás navrhla automat zcela správně a dostala plný počet bodů, ostatní dělali chyby těchto typů:
1) Pokusili se o sestavení pomocí paralelní synchronní kompozice dvou automatů, které by odpovídaly dvěma podmínkám v definici jazyka. To bohužel nešlo, FA pro slova s více 'a' než 'b' nelze vytvořit. Počet bodů dle výsledku snažení.
2) Někteří tvrdili, že automat nelze sestrojit. Lze. 0 bodů.
3) Vytvořili jste nedeterministický automat. Pokud byl jinak zcela správně, bylo to za 7-8 bodů (podle toho, jestli byl nedeterminismus jen jednou a dal se považovat za omyl).
4) Navržený automat neakceptoval některý typ slov, který akceptovat měl. Jazyk měl obsahovat slova a+, a.a+.b.a, a+.b.a+, b.a.a+, bodování zapomenutí každého typu trestalo -2 body.*
5) Automat akceptoval slova s více 'b'. Pokud to byla pouze slova s dvěma 'b', byla za to srážka 2 body. Pokud se ale objevila smyčka, která umožnila akceptovat libovolný počet 'b', mohli jste dostat max. 1 bod - takovou chybu považuji za opravdu zásadní nepochopení principu fungování automatu.
6) Další drobné chyby - automat akceptující prázdné slovo apod. - se projevily strhnutím jednoho až dvou bodů.

Podzim 2010 , 18.12.2010, Vnitro

Nechť L, R jsou jazyky nad stejnou abecedou. Rozhodněte, zda platí následující implikace. Svá rozhodnutí zdůvodněte.

Příklad 4
10 bodů

(a) $L.R$ není regulární $\implies$ L není regulární a R není regulární

(b) R je konečný $\implies$ $co-(L \cup co-R)$ je regulární

Tvrzení a) (5b): L.R není regulární => L není regulární a R není regulární.
Toto tvrzení neplatí. K tomu, aby L.R bylo neregulární, stačí neregulárnost jednoho z jazyků. Např. R={\epsilon}, L=lib. nereg. jazyk, L.R=L.
Hodnocení: Za určení, že implikace neplatí, jste získali 1 bod. Pokud jste dále pouze uvedli, že stačí neregulárnost jednoho jazyka, dostali jste celkem max. 4 body - je to pravda, ukazovalo se to na cvičení. Pro plný počet bodů bylo ale potřeba uvést nějaký protipříklad. Poměrně často jste za regulární jazyk volili prázdný jazyk, to ale není vhodný příklad, protože jeho zřetězením s čímkoli dostanete opět prázdný jazyk.

Časté chyby:
1) Není pravda, že z uzávěrových vlastností plyne R.L reg => L reg a R reg. Protipříklad viz cvičení.
2) Negace tvrzení je "L.R není regulární a zároveň (L je regulární nebo R je regulární)", nikoli "L.R není regulární a zároveň (L je regulární a R je regulární)", jak mnozí uváděli.
3) Obměna implikace v zadání zní "L je regulární nebo R je regulární => L.R je regulární" a platí právě tehdy, když platí původní tvrzení.
4) {a^n} je jazyk obsahující jedno slovo, {a^n|n>0} je jazyk regulární, pro který snadno sestrojíte automat. {a^n.b^n|n>0} není regulární.
5) Neplatí {a^n|n>0}.{b^n|n>0}={a^n.b^n|n>0}, nýbrž {a^n|n>0}.{b^n|n>0}={a^n.b^m|n>0, m>0}=a+.b+

Tvrzení b) (5b): R je konečný => co-(L sjednoceno s co-R) je regulární.
Toto tvrzení platí. R je konečný => R je regulární, co-R je ko-konečný a regulární. Sjednocení ko-konečného jazyka s lib. jazykem L dá opět ko-konečný jazyk (co-R mělo konečný doplněk, co-R s něčím sjednocené je jistě stejné nebo větší než co-R, tedy doplněk bude stejný nebo menší než R), proto co-(L sjednoceno s co-R) je jazyk konečný a tudíž regulární.
Hodnocení: Za určení, že implikace platí, jste získali 1 bod. Pokud Vaše zdůvodnění obsahovalo tvrzení, že co-(L sjednoceno s co-R) je konečné, aniž to bylo korektně zdůvodněno, dostali jste max. 3 body.

Časté chyby:
1) Dokazovaní platnosti tvrzení pomocí příkladu: toto nestačí. Příklad pouze ukazuje, že jste našli jednu dvojici jazyků takovou, že tvrzení platí, Vy ale máte rozhodnout obecně pro všechny jazyky vyhovující zadání. Klidně by se mohlo stát, že pro Váš příklad tvrzení platí, ale pro jiné ne. Toto je výrazný rozdíl oproti ukazování neplatnosti tvrzení - pro neplatnost stačí ukázat, že existuje jeden případ, který tvrzení porušuje.
2) Dokazování pomocí neplatnosti negace: někteří z Vás vzali negaci tvrzení, pomocí příkladu ukázali neplatnost negace a z toho odvozovali, že původní tvrzení platí. Takto nelze postupovat, protože není nikde řečeno, že musí platit buď tvrzení, nebo jeho negace - může se stát, že neplatí ani jedno z toho.
3) Tvrzení, že doplněk nekonečného jazyka je vždy konečný. Toto není pravda, mimo jiné by to znamenalo, že doplněk každého neregulárního jazyka je regulární.

Podzim 2010 , 18.12.2010, Vnitro

Rozhodněte, zda je jazyk $L = \{w \in \{a,b\}^* \mid \#_a(w) > \#_b(w) \lor \#_b(w) < 2\}$ regulární. Své tvrzení dokažte. *(Pro důkaz, že jazyk je regulární, stačí napsat odpovídající gramatiku nebo automat.)*

Příklad 5
15 bodů

13b

pro důkaz neregulárnosti použiju Pumping Lemma

- nechť $n \in \mathbb{N}$ je libovolné
- zvolme ~~B~~ $w \in L$ tak, aby platilo $|w| \geq n$
  např. $a^{n+1}b^n$
- ~~platí~~ všechna možná rozdělení ~~z~~ $w = xyz$, $|xy| \leq n$, $y \neq \varepsilon$

$$x = a^k \quad k \geq 0$$
$$y = a^l \quad l > 0, \quad k + l \leq n$$
$$z = a^{n+1-k-l} b^n$$

- zvolme libovolné i, tak aby ~~xy^iz~~ $xy^iz \notin L$

např. $i = 0$

$xy^iz = a^k a^{0 \cdot l} a^{n+1-k-l} b^n = a^{n+1-l} b^n$ $\notin L$ jelikož $l > 0$ a tedy $(1-l) \leq 0$ a z toho plyne, že může vzniknout nejvýše $a^n b^n$ což nesplňuje podmínku $\#_a(w) > \#_b(w)$

!!

$\Rightarrow$ jazyk L není regulární

Pozor ale, nefunguje pro $n=1$, tedy pro $\#_a(w) < 2$ a $w \in L$.

Podzim 2010 , 18.12.2010, Vnitro
Příklad 05

Za správnu odpoveď, že jazyk nie je regulárny ste získali 2 body. Dokázať ste to mohli pomocou P.L. alebo M.N. vety.

1. Pummping lemma

Tu bolo kľúčovým správne si zvoliť slovo w z jazyka L. Takým je napríklad w=b^na^(n+1). Kedy po napumpovaní, napríklad pre i=3 dostávam slovo w'=b^(n+2k)a^(n+l). Keďže platí, že k>0, tak určite #b(w')=(n+2k)>=(n+1)=#b(a) a tiež #b(w')>2. Preto w' nepatrí do L a z P.L. plynie, že jazyk L nie je regulárny.

Častou chybou bola, nesprávna voľba slova w, kedy ste po napumpovaní dostali napríklad slovo w'=a^(n-k)b^n, ktoré ale pre n=1 stále patrí do jazyka L. Niektorí sa to snažili obísť tak, že postavili n>=2, čo však nie je možné. Penalizoval som stratou 2 bodov.
Iný zase rozdelili jazyk L na dva jazyky, kde jeden bol regulárny a o druhý nie, a potom argumentovali, že z uzáverových vlastností plynie, že L nie je regulárny. Táto argumentácia nie je správna, pretože obecne neplatí, že by sme zjednotením regulárneho a neregulárneho jazyka vždy dostali neregulárny jazyk (protipríklad zaznel na cvičení). Za takéto riešenie ste mohli získať maximálne 11 bodov.
Typickými chybami bolo tiež neuvažovanie všetkých možných rozdelení (zrážka 4-5 bodov), prípadne úplne zle zvolené slovo (zisk maximálne polovice bodov).
Častým javom bolo, že nesprávne rozdelenie a pumpovanie slova od zadu. (Teda neplatilo, že |xy|<=n.) Za také riešenie som tiež dával maximálne 8 bodov.
Po jednom dvoch bodov som strhával za malé chybky (<= namiesto < a podobne), chýbajúce podmienky a podobne.

Mnohé riešenia vyzerali tak, že niekde na papieri sa "vyskytlo" slovo w, potom rozdelenia a záver, bez akéhokoľvek komentu. Je mi jasné, že dôkaz pomocou pummping lemma je notoricky známy a dá sa poňať tak, že vyplňujem šablónu, ale aj tú šablónu je nutné uviesť, aby to celé dávalo zmysel.

2. M.N. veta

Dôkaz pomocou M.N. vety sa dal viesť napríklad nasledujúcim spôsobom: Predpokladajme, že L je regulárny a teda podľa M.N. vety je index ~L konečný a tak existuje n \in N také, že |~L| = n.

Uvažujme n+1 slov:
aa, a^3, a^4, ... ,a^{n+2}

Keďže index je n a slov je n+j, tak aspoň dva musia byť v relaci ~L. Teda existujú i,j \in N, 2<i,j<=n+2, také, že a^i ~L a^j. Bez ujmy na obecnosti prepokladajme, že i<j.
Potom ale platí, že
a^ib^i nepatrí do L
a^jb^i patrí do L.
Teda z definície ~L vieme, že a^i nie je v relácii s a^j. Dostávame, že |~L| > n, čo je spor s naším predpokladom (|~L| = n) a jazyk L nie je regulárny.

Častou chybou (podobne ako pri P.L.) bolo opomenutie podmienky, že slovo patrí do L ak je počet b-čok menší ako 2. Zrážka bola tiež podobná.
Mnohí spolu plietli ~L a ~, čo svedčilo o nepochopení M.N. vety. Také riešenia som typicky hodnotil maximálne 8 bodmi.
Ďalej som strhával body za nejasné a chýbajúce argumenty.
Chcem podotknúť, že pri použití čisto Nerodovej vety, je štruktúra dôkazu podobná, ale argumentácia je iná. A teda, ak máte dve slová, ktoré sú vo ~, niečo k nim prireťazíte, pričom jedno slovo potom patrí do L a druhé nie, tak tu spor ešte nie je. Ten je až o krok ďalej a treba sa odvolať na tú časť vety, ktorá vraví o tom, že L je zjednotením niektorých tried rozkladu. Čo takmer nikto nespravil.

Ešte poznámka k riešeniam obecne. Miestami som sa pri opravovaní cítil ako jasnovidec. Teda, že by som mal vidieť na papieri aj veci, ktoré tam neboli. Je mi jasné, že ani cvičiaci na tabuľu nepíšu úplne všetko, ale zato to komentujú. Bohužiaľ, komenty na papieri nepočuť, preto ich je tam nutné písať. Alebo inak, opravujúci môže obodovať iba to, čo tam napíšete.

Podzim 2010 , 18.12.2010, Vnitro

K zadanému deterministickému konečnému automatu $\mathcal{A}$ zkonstruujte ekvivalentní minimální konečný automat v kanonickém tvaru. Konstrukci zde uveďte.

Příklad 6
12 bodů

$\mathcal{A}$	a	b
$\leftrightarrow 1$	3	6
$\leftarrow 2$	6	3
3	5	5
$\leftarrow 4$	4	5
$\leftarrow 5$	5	4
$\leftarrow 6$	-	1

1. odstraníme nedosažitelné stavy a ztotálníme automat

A'	a	b
$\leftrightarrow$ 1	3	6
3	5	5
$\leftarrow$ 4	4	5
$\leftarrow$ 5	5	4
$\leftarrow$ 6	N	1
N	N	N

2. postupně vytvoříme redukt automatu, $A/\equiv$

	$\equiv_0$	a	b
I	1	II	III
II	3	III	III
	N	II	II
III	4	III	III
	5	III	III
	6	II	I

-2

$\Rightarrow$

	$\equiv_1$	a	b
I	1	II	V
II	3	IV	IV
III	N	III	III
IV	4	IV	IV
	5	IV	IV
V	6	III	I

$\Rightarrow \equiv_1 = \equiv_2 \Rightarrow$

$A/\equiv$	a	b
$\leftrightarrow$ I	II	V
II	IV	IV
III	III	III
$\leftarrow$ IV	IV	IV
$\leftarrow$ V	III	I

$\Rightarrow$

3. provedu kanonizaci

$\Rightarrow M$	a	b
$\leftrightarrow$ A	B	C
B	D	D
$\leftarrow$ C	E	A
$\leftarrow$ D	D	D
E	E	E

A $\rightarrow$ I
B $\rightarrow$ II
C $\rightarrow$ V
D $\rightarrow$ IV
E $\rightarrow$ III

10,5

Vaším úkolem bylo zminimalizovat a zkanonizovat zadaný automat. Příklad se zdál velmi lehký, nicméně úplně správných řešení nebylo tolik, kolik jsem očekával. Příklad na jednoduchý algoritmus v sobě ukrývá pro některé z vás několik záludností. Zadaný automat nemá totální přechodovou funkci a obsahuje jeden nedosažitelný stav. Takový automat sám o sobě nemůže být vstupem algoritmu na minimalizaci DFA, jehož princip drtivá většina z vás ovládá, a je potřeba jej nejprve ztotálnit a odstranit nedosažitelný stav. Dále iniciální stav 1 je zároveň stavem akceptujícím, což bohužel zmátlo spoustu z vás.

Za příklad bylo možno získat 12 bodů, přičemž 8 jste mohli získat za správnou minimalizaci a 4 za správnou kanonizaci. Bodování bylo, myslím, poměrně vstřícné a u mnohých z vás jsem se nadřel, abych našel v písemkách věci, za které bych mohl udělit kladné body. Pokud jste uvedli správný výsledek, bodovým základem
bylo 12 bodů, ze kterých jsem odčítal body za chyby v postupu/odchýlení od dokázaného algoritmu a formální chyby. Zde je přehled nejčastějších:
Neodstranění nedosažitelného stavu na začátku algoritmu: -1b
Neodstranění nedosažitelného stavu ani ve výsledku: -4b
(takový automat jistě není minimální!)
Chybné zařazení stavu 1 mezi neakceptující,
popř. speciální třída ekvivalence pro tento stav: -2b
Neoznačení iniciálních a akceptujících stavů: -1/2b (některé/všechny)
Přepsání se v postupu bez ovlivnění výsledku: -1b
Přepsání ovlivňující výsledek: -2b
Absence definice přechodů pro nově přidaný stav: -1b

Za ostatní chyby jsem strhával body podle závažnosti "provinění". U těch z vás, kdo jste automat neztotálnili a tudíž dospěli ke špatnému výsledku, jsem dával body za věci, které jste měli dobře. Většinou odstranění nedosažitelného stavu(+1) či kanonizace(až 4b)(pozor, minimální automat i automat v kanonickém tvaru musí mít totální přechodovou funkci!). Bohužel valná část z vás dosáhla pouze na jeden bod za odstranění neakceptujícího stavu.

Podzim 2010 , 18.12.2010, Vnitro

Rozhodněte, zda existuje jazyk L nad abecedou $\Sigma = \{a, b\}$ takový, že

Příklad 7
15 bodů

$\exists$

(a) L je konečný a $a \sim_L ab$.

(b) L je nekonečný a $a \sim_L ab$.

(c) L není regulární a $\sim_L = \Sigma^* \times \Sigma^*$.

Pokud rozhodnete, že jazyk existuje, uveďte příklad takového jazyka. V opačném případě své tvrzení dokažte.

Vzhledem k tomu, že jsme s přednášejícím usoudili, že byl tento příklad o něco těžší, než bývá na vnitrosemestrálních písemkách zvykem, hodnotil jsem hodně mírně a bodoval víc než velkoryse. Dával jsem tedy spoustu bodů i za věci, za které bych normálně body nedával nebo dával bodů míň.

(a) Správným řešením je třeba prázdný jazyk. Existují i jiná správná řešení, třeba jazyk {epsilon} nebo jazyk {b} apod. Za správné řešení bylo 5 bodů.
Častá chybná řešení:
- L = {a, ab} není správné řešení, protože a.b je v L, zatímco ab.b není v L, a proto neplatí a ~L ab. Řešení podobná tomuto byla docela častá. Pokud alespoň platilo, že a i ab buď obě jsou v L nebo obě nejsou v L a L byl konečný, hodnotil jsem 3 body.
- Jiná chybná řešení (L není konečný, L obsahuje jen jedno z a, ab, apod.) jsem hodnotil 2 body.
- Pouhou odpověď ANO, která nebyla doplněna alespoň pokusem o zapsání jazyka, jsem hodnotil 1 bodem.

(b) Správný řešením byl třeba jazyk Sigma^*. Existovala i jiná správná řešení, jako {a}.{b}^* apod. Za správné řešení bylo 5 bodů.
Zde bylo o něco míň chybných řešení, ale přesto se občas vyskytovalo třeba L = {a^n b^m | n, m >= 0}. To není správně, protože a.a je v L, zatímco ab.a není v L a proto neplatí a ~L ab. Podobně jako v předchozím jsem hodnotil 3 body, pokud alespoň a i ab obě byla nebo obě nebyla v jazyce. Ostatní špatná řešení pak byla za 2 body, pokud byl alespoň pokus o napsání jazyka, samotná odpověď ANO za 1 bod.

(c) V podstatě existují dva způsoby jak ukázat, že takový jazyk neexistuje.

První způsob: ~L = Sigma^* X Sigma^* platí jen pro dva jazyky, Sigma^* a prázdný jazyk. Oba tyto jazyky jsou regulární, proto nemůže existovat žádný neregulární jazyk splňující podmínku (c).
O tento důkaz se mnozí pokusili, ale skoro všichni zapomněli na prázdný jazyk. Pokud byl jinak důkaz dobře, hodnotil jsem 4 body. Pokud chyběla ještě nějaká část zdůvodnění, hodnotil jsem 3 body. Za správnou odpověď bylo samozřejmě 5 bodů.

Druhý způsob. Podle Myhill-Nerodovy věty musí mít ~L pro neregulární jazyk L index rovný nekonečnu. Index ~L ze zadání je ovšem 1 (má jen jednu třídu). Proto nemůže existovat neregulární jazyk splňující podmínku ze zadání.
Opět bylo pokusů spousta, ne všichni ale dokázali správně zformulovat myšlenku, hodnocení bylo opět 3-5 bodů, podle toho, jak moc se (pokus o) důkaz blížil skutečnému důkazu.

Kromě toho bylo spousta více či méně zmatených a nesmyslných pokusů o důkaz, ty jsem hodnotil většinou 2 body. Samotnou odpověď NE jsem hodnotil 1 bodem.

Nakonec jsem strhával 1 bod (na list) za případné formální nedostatky (zápis jazyků apod.) a nesmyslná tvrzení. V několika málo případech (cca kolem pěti), kdy množství nesmyslů přesahovalo únosnou míru, jsem strhl 2 body.

Pár příkladů formálních nedostatků a nesmyslů
L = {a.b^*} není správný zápis jazyka.
L = {a^n b^n} rozhodně není správný zápis jazyka, navíc ani není jasné, co se tím myslí.
L = {{a,b}^*|slovo začíná na a} není správný zápis jazyka.
L = {u, v \in {a,b}^* | #a(u) = #a(v)} vůbec není zápis jazyka. Ani nechápu, co má tenhle zápis znamenat. Množinu dvojic?

V bodě (c) někteří považovali Sigma^* X Sigma^* za jazyk. Není to jazyk. Je to relace.

Z odpovědí některých studentů nebylo poznat, jestli se snaží ukázat, že L ze zadání existuje nebo naopak že neexistuje. Doporučoval bych vám na otázku typu "Rozhodněte a dokažte" vždy jasně napsat "ANO" nebo "NE", případně "existuje", "neexistuje".
Poměrně překvapivá část z vás vůbec nepochopila (nečetla?) zadání a místo aby rozhodovali o existenci jazyka, snažili se o něčem (nevím o čem) ukazovat, že to je nebo není ekvivalence. Čtěte zadání.

Podzim 2007, 10.1.2008, Termín 02
Příklad 01

Navrhněte bezkontextovou gramatiku $\mathcal{G}$ v Greibachové normální formě generující jazyk

Příklad 1
40 bodů

$$L = \{a^i b^{2j} c^i b^{2k} \mid i, j, k > 0\}.$$

(Rovnost $L = L(\mathcal{G})$ není třeba dokazovat.)

Úkolem bylo napsat gramatiku v GNF generujici jazyk {$a^i b^{2k} c^i b^{2k}$ | i,j,k > 0}. Chyby, ktere jste delali, se daji rozdelit do dvou skupin:
1)Gramatika generovala spatny jazyk - za toto jsem strhaval az 25 bodu (resp. az 40 pokud se jazyk generovany gramatikou ani vzdalene nepodobal zadani). Nejcasteji gramatika negenerovala stejny pocet "a" a "c", pripadne generovala lichy pocet "b". Nekteri z vas take zahrnuli moznost "i,j,k ≥ 0".
2)Gramatika nebyla v GNF - i tady jsem strhaval az 25 bodu. Druhy chyb se ruznily. Nekteri napsali gramatiku, ktera nejevila znamky zadne normalni formy. Ti snazivejsi napsali gramatiku v CNF, pripadne napsali gramatiku "temer" v GNF (obsahovala ε-pravidla, pripadne nejaka prava strana pravidla zacinala neterminalem).

```
S -> AB,
A -> aAc / aBc,
B -> bb / bbB

v GNF
S -> aA[c]B / aB[c]B,
A -> aA[c] / aB[c],
B -> b[b] / b[b]B,
[b] -> b,
[c] -> c
```

Podzim 2007, 10.1.2008, Termín 02
Příklad 02

Příklad 2
33 bodů

Uvažme následující relace na slovech nad abecedou $\Sigma = \{a, b\}$. U každé relace určete, zda se jedná o pravou kongruenci. Pokud rozhodnete, že se o pravou kongruenci nejedná, dokažte to. V opačném případě určete index ekvivalence a popište jednotlivé třídy ekvivalence.

(a) $u \sim v \overset{def}{\iff} |u| \leq |v|$

(b) $u \sim v \overset{def}{\iff} u, v \in \{a\}.\Sigma^+ \text{ nebo } u, v \in \{b\}.\Sigma^+ \text{ nebo } u = v$

Reseni:
a) Relace neni pr. kongruence, protoze neni symetricka (tudiz ani ekvivalence).
b) Relace je pr. kongruence, index ekvivalence je 5 a jednotlive tridy nasledujici: {\eps}, {a}, {b}, {a}{a,b}^+, {b}{a,b}^+.

Bodovani:
a) Tady to bylo narocne, drtiva vetsina z vas overeni relace zda je ekvivalenci vynechala. Proto jsem se i u spatnych reseni snazila hodnotit alespon overeni podminky priretezeni zprava, a jakekoli naznaky, ze problematice rozumite :).
b) S timto prikladem mela vetsina z vas take problemy - hlavne se zapisem trid ekvivalence, do kterych se snazili dostat slova u, v a jejich vzajemny vztah (jako napr trida, kdy u = v, nebo jeste lepe u != v). Nic takoveho sem nepatri, a spis vam to uskodilo :|.

Podzim 2007, 10.1.2008, Termín 02
Příklad 03

Je dána gramatika $G = (\{S, A, B\}, \{a, b, c\}, P, S)$, kde

Příklad 3
20+15 bodů

$$
\begin{aligned}
P = \{\ S &\rightarrow AaB \mid Bb \mid \varepsilon \\
A &\rightarrow cAb \mid aS \mid Ab \\
B &\rightarrow Bb \mid bA\ \}.
\end{aligned}
$$

(a) Zkonstruujte PDA $\mathcal{A}$ pro nedeterministickou syntaktickou analýzu shora dolů. Uveďte způsob akceptování.

(b) Zapište akceptující výpočet automatu $\mathcal{A}$ nad slovem $bcabb$.

S timto prikladem jste si vetsinou poradili velmi dobre. Bodovani bylo nasledujici:

a) 20 bodu (2 body za sedmici, 2 body za spravny zpusob akceptovani, 16 za prechodovou funkci)
Typicke chyby:
- velmi mnoho z Vas melo problem se sedmici - zasobnikovy automat neni plnne urcen pouze svou prechodovou funkci a sedmice je jeho nedilnou soucasti
- chybejici zpusob akceptovani - pokud nebyla uvedena sedmice, ani zpusob akceptovani -2 body, pokud byla uvedena sedmice a z ni byla citelna mnozina akceptujicich stavu, ale nebyl explicitne uveden zpusob akceptovani -1 bod
- je dulezite vedet, jaky je typ prechodove funkce, mnohdy jste zapominali na mnozinove zavorky (-3 body), zapominali je u jednoprvkovych mnozin (-1 bod), nebo funkci nenapsali jako funkci, ale jako relaci - napr. misto delta(q,\eps,S)={(q,\eps),(q,Bb)} jste psali delta(q,\eps,S)=(q,\eps), delta(q,\eps,S)=(q,Bb) (-4 body). Toto nejsou pouze formality, jde o porozumeni/neporozumeni pojmu prechodova funkce.
- formalni chybky, napr. -> misto =, zapominani znaku \delta, atd. (-1 bod)
- casto Vase reseni neodpovidala algoritmu pro analyzu shora dolu. Tady se bodove srazky pohybovaly okolo -1 az -5 bodu, podle zavaznosti a vlivu na funkci PDA.

b) 15 bodu
Caste chyby:
- formalni nedostatky - u konfiguraci navic symbol \delta, misto |- pisete ->,...(-1 bod)
- zacinani s jinym nez S symbolem na zasobniku -2 body
- chybejici vstupni slovo v konfiguraci -2 body
- uvedeni pouze odvozeni z gramatiky, celkem 1 bod za celou odrazku b)

Podzim 2007, 10.1.2008, Termín 02
Příklad 04

Zformulujte algoritmus, který k dané bezkontextové gramatice zkonstruuje jazykově ekvivalentní gramatiku bez ε-pravidel.

Příklad 4
40 bodů

Správné řešení: Algoritmus na odstraňování epsilon pravidel z bezkontextové gramatiky, například ten, co je ve skriptech či na slajdech.

Bodování:
Tři hlavní části algoritmu jsou

a) napočítání množiny N_epsilon (bez toho to nejde, dojde k zacyklení apod.)
- 12 bodů
- typické bodové srážky:
** jen neterminály, které se na epsilon přepíšou v jediném kroku: -6b*
** jen neterminály, které se na epsilon přepíšou sice v libovolném počtu kroků, ale jen skrze jednoduchá pravidla: -4b*
** jenom příklad, žádný rozumný popis: -9b*

b) vytváření nových pravidel
- 18 bodů
- typické bodové srážky:
** nezachovají se původní pravidla: -4b*
** chybí (nebo je nejasně napsáno), že se vynechává libovolná kombinace neterminálů z N_epsilon: -4b*
** nevyhodí se epsilon pravidla, nebo se přidají nová: -6b*
** pokud nebylo popsáno nějak slušně, ale jen příkladem, max počet bodů byl 10 místo 18 plus většinou ještě nějaká srážka*

c) když v původní gramatice S => epsilon, v nové je třeba přidat S' -> S | eps*
- 10 bodů
- většinou 10 nebo nic, srážky za nedostatečné slovní vysvětlení, kdy a proč - až 5 bodů

Tento výčet nezahrnuje velkou spoustu velmi specifických a unikátních chyb (snažil jsem se vždy vysvětlit na papír s příkladem). Rovněž jsem srážel jeden až dva body za drobné formální nedostatky (špatné závorky, když někdo nazývá neterminály jinak (stavy, body) apod.) Nulou bodů pak byly hodnoceny zcela fatální chyby, jako zcela jiný algoritmus (CNF, odstranění jednoduchých pravidel), algoritmus pracující s konečným automatem apod.

Podzim 2007, 10.1.2008, Termín 02
Příklad 05

Nechť L je bezkontextový jazyk. Dokažte nebo vyvraťte následující implikace:

Příklad 5
16+16 bodů

(a) L má vlastnost sebevložení $\implies$ L je nekonečný

(b) L je nekonečný $\implies$ L má vlastnost sebevložení

Reseni:
Protoze pro L z CFL plati:
L je regularni <=> nema vlastnost sebevlozeni,
tak jiste kazdy jazyk s vlastnosti sebevlozeni je neregularni a tudiz neni konecny. Vsechny konecne jsou totiz regularni. Prvni uvedena implikace ze zadani tedy plati, vlastnost sebevlozeni implikuje nekonecnost.
Naopak ale staci uvazit, ze existuji nekonecne regularni jazyky. Tak napriklad jazyk popsany gramatikou
G=({S},{a},{S->a|aS},S)
je nekonecny a regularni, tudiz (zde navic zrejme) nema vlastnost sebevlozeni. Proto druha uvedena implikace neplati, nekonecnost neimplikuje vlastnost sebevlozeni.

Bodovani:
U kazde implikace byly 4 body za spravnou odpoved a az 12 za dukaz. Celkem tedy 32 bodu.
Casto jsem strhaval body za dukaz prvni implikace, v nemz jste pouzili existenci odvozovaci posloupnosti A=>+uAv, ale zanedbali jste dulezity fakt, a to, ze potrebujete, aby A byl _pouzitelny_ neterminal. To je opravdu dulezite, protoze jinak by bylo mozne dokazat, ze kazdy jazyk je nekonecny.
Ve druhe implikaci Vam stacilo uvest jazyk, ktery je protiprikladem jeji platnosti a nejakym zpusobem dokazat, ze nema vlastnost sebevlozeni a je nekonecny. Regularni gramatika byla OK. Nekteri z vas uvedli vhodny jazyk, ale k nemu gramatiku se sebevlozenim, i ti dostali docela hodne bodu. Kdo ale netrefil jazyk, dostal jen ty 4.

Podzim 2007, 10.1.2008, Termín 02
Příklad 06

Příklad 6
40 bodů

Nechť $\mathcal{A} = (Q, \Sigma, \delta, q_0, F)$ je deterministický konečný automat rozpoznávající jazyk L. Označme $n = card(Q)$. Dokažte, že L je nekonečný, právě když $\mathcal{A}$ akceptuje alespoň jedno slovo w s vlastností $n \leq |w| < 2n$.

Správné řešení:
Měla se dokázat ekvivalence L je nekonečný <=> A akceptuje slovo w s vlastností n <= |w| < 2n.

Implikace A akceptuje slovo... => L je nekonecny:
Akceptuje-li A slovo w s vlastnosti n <= |w|, musel A pri jeho akceptovani projit dvakrat stejnym stavem. Tedy w = xyz, kde y neni epsilon a soucasne delta^(q_0, x) = q_1, delta^(q_1, y) = q_1 a delta^(q_1, z) = q_2, kde q_0 je pocatecni stav a q_2 je koncovy. Potom A akceptuje vsechna slova tvaru x(y^i)z, kde i je lib. prirozene cislo, tedy L nekonecny.

Implikace L je nekonecny => A akceptuje slovo...:
Je-li L nekonceny, jiste A akceptuje slovo w s vlastnosti n <= |w|. Zvolime takove slovo w, aby jeho delka byla nejmensi mozna. Je-li |w| < 2n, jsme hotovi. Je-li |w| >= 2n, jde w podobne jako v prvnim pripade rozdelit na xyz tak, ze |xy| <= n a y neni epsilon a navic i slovo xz je akceptovano automatem. Ale n <= |xz| < |w|, coz neni mozne, protoze w bylo vybrano jako nejkratsi slovo delsi nez n akceptovane automatem, takze |w| < 2n a w je nase hledane slovo.

Bodování (celkem bylo možné získat 40 bodů):
Drtivá většina z vás dokazovala pouze první implikaci, za kterou jste mohli získat až 20 bodů. Bral jsem i důkaz pumping lemmatem, i když to mluví pouze o existenci nějakého n, nikoliv o tom, že za n lze vzít kardinalitu množiny stavů, to se vyskytuje až v důkazu PL.
Kdo napsal, že automat musí při akceptování slova delšího jak n projít dvakrát stejným stavem a pořádně nezdůvodnil, co z toho vyplývá, měl 10 nebo 15 bodů, podle toho, jak moc to zdůvodnění bylo nedostatečné.
Kdo platnost implikace ukázal pouze na příkladu, měl 5 bodů, ale to pouze v případě, že ten příklad skutečně o něčem vypovídal.
Další nedostatečně obecná řešení (např. tvrzení, že musí vzniknou smyčka nad jedním stavem, apod.) byla hodnocena 5 nebo 10 body, podle toho, o jak moc velkou újmu na obecnosti šlo.

Za druhou implikaci bylo zbylých 20 bodů.
Jako důkaz druhé implikace jsem bral i opakované použití pumping lemmatu, tedy pumpování slova delšího než n "dolů" (s pumpovací konstantou i = 0), dokud nebude kratší než 2n, což by bylo potřeba formalizovat indukcí.
Těch, kteří se o tu druhou implikaci pokoušeli, moc nebylo, takže jsem jejich řešení posuzoval individuálně, proto tady nepíšu žádné konkrétnější bodování.

Podzim 2007, 20.1.2008, Termín 03
Příklad 01

Příklad 1
45 bodů

O každém z následujících jazyků rozhodněte, zda je bezkontextový. Svá tvrzení dokažte. *(Pro důkaz, že jazyk je bezkontextový, stačí napsat odpovídající gramatiku nebo automat.)*

(a) $L_1 = \{a^i b^j c^k \mid i, j, k \geq 0,\ i \neq j\}$

(b) $L_2 = \{a\}^*.\{b\}^*.\{c\}^* \smallsetminus \{a^i b^i c^i \mid i \geq 0\}$

????

Podzim 2007, 20.1.2008, Termín 03
Příklad 02

Příklad 2
30 bodů

Pomocí Myhill-Nerodovy věty dokažte, že jazyk

$$L = \{b\}.\{ww \mid w \in \{a,b\}^*\}$$

není regulární.

Vasim ukolem v prikladu 2 bylo dokazat za pouziti M.N. vety, ze jazyk:
L = {b}.{ww | w nalezi {a,b}*} neni regularni.

Bohuzel drtiva vetsina z Vas mela s timto prikladem velke potize, proto zde uvedu, jak by mohlo vypadat mozne reseni.

Predpokladejme, ze jazyk L je regularni, a tudiz z M.N vety plyne, ze index relace ~L je konecny, tudiz existuje n \in N takove, ze index ~L je roven n.
Nyni uvazme posloupnost n+1 (ruznych) slov:
bab, baab, ... ,ba^{n+1}b
Jelikoz index ~L je n, musi v teto posloupnosti existovat aspon dve (ruzna) slova takova, ze budou v jedne tride rozkladu Sigma*/~L (budou v relaci ~L).
Nyni ukazeme, ze takova slova neexistuji (tim dostaneme spor s predpokladem ze ~L ma index n).
Pro libovolna dve ruzna slova (i != j) plati:
ba^{i}b . a^{i}b = ba^{i}ba^{i}b \in L
ba^{j}b . a^{i}b = ba^{j}ba^{i}b \not \in L
Tudiz (z definice prefixove ekvivalence) slova ba^{i}b a ba^{j}b nejsou v relaci ~L a tim dostavame spor s predpokladem, ze L ma index n, a tedy dostavame, ze jazyk L neni regularni.

Bodovani prikladu 2 bylo nasledovne:
Maximalni pocet bodu byl 30.
Pokud jste uvedli pouze spravne zneni M.N. vety, dostali jste 5 bodu.
Pokud jste jasne uvedli jen to, ze je treba dokazat, ze index relace ~L je nekonecno, dostali jste 5 bodu
Pokud jste uvedli pouze obe vyse uvedene veci, dostali jste 7 bodu.
Dale se ve Vasich dukazech objevovaly mene i vice zavazne nedostaky, ktere jsem se snazil uvest do hodnoceni, a strhaval jsem za ne prislusny pocet bodu.

Nakonec jeste jednu poznamku. Hodne z vas tvrdilo, ze zretezeni regularniho jazyka s neregularnim jazykem je vzdy jazyk neregularni. Toto samozrejme neni pravda, napr. uvazme zretezeni {a}.{a^{p} | kde p je prvocislo} = {a^{i} | kde i>= 2} a to je zrejme regularni jazyk.*

Podzim 2007, 20.1.2008, Termín 03
Příklad 03

Příklad 3
22+18 bodů

Je dána gramatika $G = (\{S, A, B\}, \{a, b, c\}, P, S)$, kde

$$P = \{ \begin{array}{lcl} S & \to & AaB \mid Bb \mid \varepsilon \\ A & \to & cAb \mid bS \mid a \\ B & \to & bS \mid bc \ \}. \end{array}$$

(a) Zkonstruujte rozšířený PDA $\mathcal{A}$ pro nedeterministickou syntaktickou analýzu zdola nahoru. Uveďte způsob akceptování. Popište rozdíl mezi použitou a standardní notací.

(b) Zapište akceptující výpočet automatu $\mathcal{A}$ nad slovem $cbbbb$.

Nejcasteji jste nenapsali \delta(q, \eps, bS) = {(q,A),(q,B)}, za coz jsem strhavala 1 bod.
Dalsi srazky:
-2b za nenapsani faktu, ze vrchol zasobniku se pise vpravo (to byl ten rozdil proti standardni notaci)
-3b za spatne akceptovani
-1 az -3 body za spatne zapsany vypocet automatu
-5b za \delta(q, x, x) = {(q, \eps) } kde x je terminal (terminaly patri na zasobnik, "neumazavaji" se)
-1b za chybejici sedmici vaseho zasobnikoveho automatu

Podzim 2007, 20.1.2008, Termín 03
Příklad 04

Zformulujte algoritmus, který k danému nedeterministickému konečnému automatu $\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$ zkonstruuje jazykově ekvivalentní deterministický automat bez nedosažitelných stavů a s totální přechodovou funkcí. *(Nezapomeňte přesně popsat výstupní automat.)*

Příklad 4
40 bodů

Ad bodovani 4. prikladu:

duraz byl kladen na nekolik veci a k nim proto nekolik poznamek:
- casto jste napsali, ze z pocatecniho stavu vedeme prechody novych stavu, ktere jsou mnozinami vsech nasledniku pod stejnymi pismeny, a ze takto postupujeme i dale. Malokdo uz pak napsal, jak vypada prechodova funkce z novych stavu, tj. sjednoceni pes vsechny stavy v tomto novem stavu. Vetsinou byla vase "sjednoceni" jen souborem nasledniku jednoho puvodniho stavu.
- slovni popis byl akceptovan, ale casto byl natolik vagni, ze nebylo jasne co delat a pro co vsechno.
- vyznamna cast z vas psala o odstranovani nedosazitelnych stavu (dokonce mnozi i o neterminalech v CFG) a o ztotalneni. Ti, kteri psali o determinizaci, casto uvadeli priklad, ktery ani moc nezobecnili. Akceptoval jsem i popis pomoci tabulky (jako ve cviceni) typu "tohle dam sem a tohle trochu jinak sem", ale muselo byt jasne, co delate.
- vetsina se prilis nezatezovala koncovymi stavy noveho automatu
- akceptoval jsem plne dokonce i reseni naucena nazpamet (obdivuji vasi pamet), u kterych bylo celkem zrejme, ze vubec nevite co pisete (U p \in M ... napsano v radku, protoze ve skriptech je to nikoli pod sjednoceni, ale jako dolni index, takze ani nebylo videt, co sjednocujete)
- znacna cast bodu hodnotila aspon naznaky nejakeho postupu (zjm. jakykoli pokus o podmnozinovou konstrukci). Body jste dostavali, i kdyz to bylo treba hodne spatne, ale byl tam aspon naznak porozumeni. Pokud mate k hodnoceni prikladu nejake vyhrady, muzete mi napsat mail, subject: IB102, prilozte pisemku a ja vam zkusim odpovedet co nejrychleji (ale od zitrejsiho vecera do nedele vcetne jsem bohuzel mimo mail)

Podzim 2007, 20.1.2008, Termín 03
Příklad 05

Nechť L, R jsou jazyky nad abecedou $\{a, b\}$. Dokažte nebo vyvraťte následující implikace:

Příklad 5
15+15 bodů

(a) L a R jsou neregulární $\implies$ $L \cap \text{co}-R$ není regulární

(b) L a R jsou deterministické bezkontextové $\implies$ $L \cup \text{co}-R$ je bezkontextový

Za ukol bylo dokazat (ne)platnost nasledujicich tvrzeni
(a)L a R jsou neregularni => L∩R neni regularni
(b)L a R nejsou deterministicke bezkontextove => L∪R je bezkontextovy

ad (a): neplati, stacilo vzit za L libovolny neregularni a polozit R=L
ab (b): platí, pro dukaz stacilo, ze trida determ. bezkontextove jazyky je uzavrena na doplnek a ze trida bezkontextovych jazyku je uzavrena na sjednoceni. Nejcastejsi chybou bylo tvrzeni, ze nektera ze trid je uzavrena na obe operace.

Podzim 2007, 20.1.2008, Termín 03
Příklad 06

Nechť $\mathcal{M} = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ je PDA.

Příklad 6
7+5+9+14 bodů

(a) Napište typ funkce δ.

(b) Definujte *konfiguraci* PDA.

(c) Definujte relaci *krok výpočtu* ($\vdash_{\mathcal{M}}$).

(d) Napište podmínky, které musí PDA splňovat, aby byl *deterministický*.

M = (Q, \Sigma, \Gama, delta, q_0, Z_0, F)

a) definovat typ prechodove funkce PDA. Spravne reseni:
delta: Q x (\Sigma sjednoceno {\epsilon}) x \Gama -> P_fin(Q x \Gama^*)

b) definovat konfiguraci PDA. Spravne reseni:
konfigurace je prvek mnoziny Q x \Sigma^* x \Gama^*. Prvni komponenta je stav PDA, druha je jeste neprectena cast vstupniho slova a treti je cely obsah zasobniku s vrcholem vlevo.

c) Definovat krok vypoctu PDA.
(q, aw, Z.\alpha) |- (r, w, \beta.\alpha) <=> (r, \beta) je prvkem delta(q, a, Z),
kde q, r jsou prvkem Q,
a je prvkem \Sigma sjednoceno {\epsilon}.
w je prvkem \Sigma^*.
Z je prvkem \Gama,
\alpha je prvkem \Gama^*

d) Napsat podminky, za kterych je PDA deterministicky. Spravne reseni:
i) Pro kazde q z Q a Z z \Gama, pokud delta(q, \epsilon, Z) neni prazdna mnozina, tak delta(q, a, Z) je prazdna mnozina pro kazde a ze \Sigma
ii) Pro kazde q z Q, a ze \Sigma sjednoceno {\epsilon} a Z z \Gama, obsahuje mnozina delta(q, a, Z) maximalne jeden prvek.

Bodovani:
a) (max. 7 bodu) Nejcastejsi chybou bylo, ze jste zapomneli na P_fin a jako vystupni typ funkce jste napsali pouze Q x \Gama^*. Za to jsem daval 5 bodu. Dale jste dost casto pridavali nebo naopak odebirali hvezdicky jednotlivym komponentam, za to jsem strhaval 1-2 body podle zavaznosti.

b) (max. 5 bodu) V konfiguraci jste casto misto celeho neprecteneho vstupu a celeho zasobniku uvadeli pouze prvni symbol vstupu a vrchol zasobniku. Za to jsem daval 1-2 body podle toho, jak to bylo napsano. Pokud jste zapomneli, ze soucasti konfigurace je vstup, dostali jste 2-3 body, opet podle toho, jak to bylo napsano.

c) (max. 9 bodu) Pokud jste nevedeli, co je to konfigurace, bohuzel jste nemohli vyresit tuhle ulohu spravne. Nicmene za nejake nadejne "pokusy" tohoto typu jsem dal 1-2 body. Hodne z Vas zvolilo slovni popis, ktery byl bohuzel vetsinou dost neurcity, takze jsem i za ty lepsi daval pouze 3-6 bodu.

d) (max. 14 bodu) Podminky jsou dve a za kazdou bylo 7 bodu. Nejcasteji jste zapominali podminku (i), cimz jste o 7 bodu prisli. Spousta z Vas opet zvolila slovni popis, ktery byl casto velmi neurcity a daval jsem za nej misto 7 2-5 bodu.

Podzim 2007, 29.1.2008, Termín 04
Příklad 01

Navrhněte zásobníkový automat akceptující jazyk

Příklad 1
40 bodů

$$L = \{w.c^n \mid w \in \{a,b\}^*,\ n \geq 0,\ \#_a(w) > n\}.$$

Uveďte, jakým způsobem navržený automat akceptuje.

Reseni mohlo vypadat napr. takto:

M = ({q_0, q_1, q_2}, {a, b, c}, {A, Z}, delta, q_0, Z, {q_2})

delta(q_0, a, Z) = {(q_0, AZ)}
delta(q_0, a, A) = {(q_0, AA)}
delta(q_0, b, Z) = {(q_0, Z)}
delta(q_0, b, A) = {(q_0, A)}
delta(q_0, \epsilon, A) = {(q_1, A)}
delta(q_1, c, A) = {(q_1, \epsilon)}
delta(q_1, \epsilon, A) = {(q_2, A)}

PDA M akceptuje koncovym stavem.

Bodovani (max. 40b)
Bodove srazky:
-Automat umoznoval libovolne stridat a, b, c ... -20b
-Automat neumoznoval stridat a, b ... -20b
-Automat akceptuje pouze "malou" podmnozinu L (napr. pouze slova w.c^n takova ze #_a(w) = n + 1) ... -20b
-Spatne osetrene "krajni pripady" (automat akceptoval nejaka "spatna slova", napr. b, \epsilon, ac, atd.; nebo naopak automat neakceptoval nejaka "dobra slova", nejcastejsi bylo, ze automat neakceptoval slova, kde nebylo ani jedno c) ... -5b
-Pouzivani notace rozsireneho zasobnikoveho automatu bez uvedeni, ze jde o rozsireny automat ... -5b
-Nevyznaceni koncoveho stavu ... -5b

Podobne jsem strhaval 5b i za dalsi mene zavazne chyby. Pokud bylo chyb vice a nejakym zpusobem spolu souvisely, nestrhaval jsem vzdy 5b za kazdou, ale nekdy jsem strhl za dve chyby pouze 5b.

Nektere spec. pripady:
-Pokud nekdo navrhoval gramatiku misto automatu, dostal celkem 10b, pokud byla gramatika dobre
-Pokud nekdo napsal pouze myslenku (Kdyz budu cist a-cka, budu pridavat symboly na zasobnik, kdyz budu cist b-cka, nebudu se zasobnikem delat nic. Kdyz budu cist c-cka, prejdu do jineho stavu, abych uz nemohl cist a-cka a b-cka, v tomto stavu budu pri cteni c-cek odebirat symboly ze zasobniku, pokud tam jeste nejake budou dovolim akceptovat) dostal maximalne 20b
-Pokud nekdo problem vyresil automatem s nekonecne mnoha stavy, dostal 5b

Podzim 2007, 29.1.2008, Termín 04
Příklad 02

Příklad 2
33 bodů

Pomocí Myhill-Nerodovy věty dokažte, že jazyk

$$L = \{a^i b^j c^k \mid i, j, k > 0,\ j \neq k\}$$

není regulární.

Bohuzel drtiva vetsina z Vas mela opet s timto prikladem velke potize, prestoze velice podobny priklad byl v minulem terminu a v D.F. jsem napsal jak by mohlo vypadat spravne reseni.

Bodovani prikladu 2 bylo nasledovne:
Maximalni pocet bodu byl 33.
Pokud jste uvedli pouze spravne zneni M.N. vety, dostali jste 5 bodu.
Pokud jste jasne uvedli jen to, ze je treba dokazat, ze index relace ~L je nekonecno, dostali jste 5 bodu
Pokud jste uvedli pouze obe vyse uvedene veci, dostali jste 7-8 bodu.
Dale se ve Vasich dukazech objevovaly mene i vice zavazne nedostaky, ktere jsem se snazil uvest do hodnoceni, a strhaval jsem za ne prislusny pocet bodu.

Podzim 2007, 29.1.2008, Termín 04
Příklad 03

Příklad 3
45 bodů

Je dána bezkontextová gramatika $\mathcal{G} = (\{S, A, B\}, \{a, b, c\}, P, S)$, kde

$$
\begin{aligned}
P = \{ \; S &\rightarrow Ac \\
A &\rightarrow Ba \mid Sa \\
B &\rightarrow SS \mid b \; \}.
\end{aligned}
$$

Převeďte gramatiku $\mathcal{G}$ do Greibachové normální formy. Použijete-li algoritmus z přednášky, uveďte zvolené uspořádání neterminálů. V ostatních případech dokažte, že výsledná gramatika je jazykově ekvivalentní gramatice ze zadání.

Bodovani bylo nasledujici:
meli jste prevest zadanou gramatiku do GNF, coz melo dve casti -- odstraneni leve rekurze a prevedeni nelevorekurzivni gramatiky do GNF. Za prvni cast jsem davala 25 bodu, za tu druhou 20 bodu, pricemz jsem tyto dve ulohy brala jako samostatne, tedy pokud jste neodstranili dobre levou rekurzi, mohli jste porad ziskat 20 bodu za prevod Vami vytvorene gramatiky do GNF. V tomto pripade jsem Vam tam vetsinou psala poznamku: "prevod spravne, pravidla spatne" nebo neco v tom duchu. Opacny pripad, tedy spravne odstraneni leve rekurze a nespravne prevedeni do GNF byl ohodnocem 25 body (pokud odstraneni rekurze bylo v poradku).

Pokud se Vam aspon castecne darilo levou rekurzi odstranit, ale nepodarilo se Vam to u vsech pravidel, tak za kazde pravidlo spravne bylo 5 bodu, pokud jste u nej navic i odstranili bezprostredni levou rekurzi, bylo za to dalsich 5 bodu. Bohuzel, casto pokud se nekdo zamotal do prikladu hned zezacatku, tak se pak jiz k nejakemu vetsimu zisku nedostal. Pouze v pripade, ze jste udelali jednou chybu a ta se s vami "vezla" cely priklad, pak jsem to Vam strhavala body jen za tu jednu chybu.

Vyskytly se take pripady, kdy jste nepouzili algoritmus z prednasky. V tom pripade, jak bylo napsano i v zadani, jste meli _dokazat_, ze Vami vytvorena gramatika je ekvivalentni s gramatikou puvodni. Nikdo z tech, koho se to tyka, tak neucinil, ziskal tedy 0 bodu.

V nekterych pripadech bylo reseni tak zamotane, ze jsem ani pres veskerou snahu nepochopila, co se snazite udelat. V takovem pripade byste tam meli mit poznamku "nechapu vase reseni" a citite-li se ukrivdeni, napiste mi mail, kde se mi pokusite Vas postup vylozit. Pozor ale, pokud ten postup nebude vyplyvat ciste z toho a jen z toho, co mate napsane v pisemce, body pridavat nebudu. (Takze maily typu "ja jsem jen zapomnel na tohle pravidlo" se velmi pravdepodobne s uspechem nestkaji!)

Nasleduje prehled srazek:
- chybejici ctverice G = (...) ... 1b
** cast odstraneni leve rekurze*
- za spatne pouziti lemmatu o substituci ... az 10 b
- za spatne odstraneni bezprostredni leve rekurze ... 10 b
- za spatnou substituci za neterminal, nebo za substituci, ktera tam vubec byt nemela ... 10 b
- za "odstraneni" jednoduchych pravidel (ac gramatika ze zadani zadna jednoducha pravidla neobsahovala, presto si to mnozi neuvedomili!) ... 10 b
** cast prevod do GNF*
- neterminal navic ... 1 b
- vytvoreni neterminalu, ke kterym jste nevytvorili pravidla ... az 15 b dle zavaznosti

Podzim 2007, 29.1.2008, Termín 04
Příklad 04

Příklad 4
40 bodů

Zformulujte algoritmus, který k dané bezkontextové gramatice $\mathcal{G} = (N, \Sigma, P, S)$ bez ε-pravidel zkonstruuje jazykově ekvivalentní gramatiku bez jednoduchých pravidel a bez ε-pravidel. *(Nezapomeňte přesně popsat výstupní gramatiku.)*

?????

Podzim 2007, 29.1.2008, Termín 04
Příklad 05

Dokažte nebo vyvraťte následující tvrzení:

Příklad 5
16+16 bodů

(a) Každá redukovaná bezkontextová gramatika generuje neprázdný jazyk.

(b) Třída regulárních jazyků je uzavřená na *průnik s bezkontextovým jazykem.*

Řešení:
Jak již psal doktor Strejček, prázdný jazyk nelze vygenerovat redukovanou gramatikou. Důkaz:
Redukovaná gramatika obsahuje aspoň iniciální neterminál S (z definice gramatiky). Ten je normovaný (z definice redukovanosti). Tedy existuje terminální řetězec vygenerovatelný z S, tento řetězec patří do jazyka gramatiky.

Průnik regulárního a bezkontextového jazyka nemusí nutně být regulární. Příklad:
R buď (regulární) jazyk všech slov nad abecedou {a,b}, L buď známý bezkontextový {a^n.b^n | n>0}. Jejich průnik je opět L, a o tom umíme dokázat, že je neregulární.

Nejčastější chybou v bodě a) bylo uvedení neredukované gramatiky pro prázdný jazyk, případně záměna prázdného jazyka s {epsilon}.
Ve druhém příkladě jste občas chybili v tom, že jste vybrali jazyk L, který nebyl ani CFL, nebo naopak byl regulární.

Podzim 2007, 29.1.2008, Termín 04
Příklad 06

Příklad 6
20+10 bodů

(a) Definujte pojem *gramatika*.

(b) Definujte, kdy se jazyk nazývá *rekursivně spočetný*.

(a) Definujte pojem _gramatika_. 20 bodů
Správné řešení:
Gramatika je čtveřice (N, Sigma, P, S) [4b], kde
N je neprázdná [1b] konečná [1b] množina neterminálů [2b],
Sigma je konečná [1b] množina terminálů [2b] taková, že N průnik Sigma = prázdná množina [1b],
P podmnožina V*NV* X V* (kde V = (N u Sigma)) [2b] je konečná [1b] množina pravidel [2b],
S je počáteční [2b] neterminál, tj. S \in N [1b].

Poznámky k hodnocení:
- když někdo u P nebo Sigma napsal "neprázdná", nestrhával jsem body
- nevyžadoval jsem "S \in N", stačilo jakkoli naznačit, že S je neterminál; pokud jsem našel jen "S je počáteční symbol" nebo snad "S je počáteční stav", ten jeden bod jsem strhnul
- pokud u P někdo napsal, že je to přechodová funkce, strhnul jsem celých 5 bodů
- když někdo napsal množinovou definici P alespoň zhruba správně, strhl jsem jen jeden bod místo dvou.
- další bodové srážky za formální nepřesnosti a specifické chyby (např., že S je množina neterminálů)

Komentář: většina zapomínala na přívlastky "neprázdná", "konečná", podmínku disjunktnosti N a Sigma, definici množiny pravidel. I bez těchto věcí jste dostali 13 (4 + 2 + 2 + 2 + 3) bodů.

Perlička: "Gramatika je pětice, G = (N, Sigma, P, S)."

(b) Definujte, kdy se jazyk nazývá _rekurzivně spočetný_. 10 bodů
Řešení: Dvě ekvivalentní možnosti. Stačila (samozřejmě) jedna.
Jazyk je rekurzivně spočetný, pokud
- existuje Turingův stroj, který jej akceptuje.
- existuje gramatika (typu 0), která jej generuje.

Hodnocení: Za úplně správnou odpověď 10 bodů, za drobné chyby ve vyjadřování v jinak správné odpovědi (TS, který "generuje" apod.), bod či dva dolů. Za definici rekurzivního jazyka místo rekurzivně spočetného (tj. "úplný TS"), byla-li jinak úplně dobře, 5 bodů. Za nesmysly 0 bodů, jen v několika výjimečných případech 1 bod, kdy to, co jsem četl, bylo alespoň pravdivé tvrzení, když už ne definice.
Komentář: Děs a hrůza. Přitom nechápu, co už byste chtěli jednoduššího než jednoduchou definici. Mnozí evidentně neměli vůbec tušení, o čem je řeč (někteří nejspíše dokonce netuší, co je to "jazyk") a psali, co je zrovna napadlo. Několik lidí psalo definici rekurzivní gramatiky. Mnozí (až překvapivě mnozí) tvrdili, že "rekurzivně spočetný" je totéž, co "konečný". Jiní zase sice netušili, co znamená "rekurzivně spočetný", ale znali slovo "rekurze" a slovo "spočítat" a snažili se to z toho nějak uvařit.
Dovolil jsem si vybrat několik ukázkových perel:

Jazyk je rekurzivně spočetný, když
...se zanořuje sám do sebe, ale v určité chvíli se zanořování zastaví.
...neobsahuje nepoužitelné neterminály typu I a typu II.
...je to jazyk, který se většinou pro velký počet nehodí zapsat jeho výčtem, ale raději jiným způsobem, ale přesto je konečný,
...právě tehdy, když je konečný = je regulární.
...jsme schopni spočítat počet rekurzí v jazyku.
...tento jazyk je vytvořený gramatikou, která obsahuje rekurzi a zároveň dokážeme spočítat počet slov v jazyce.
A na závěr: ...pokud existuje úplný Turingův stroj.

Podzim 2007, 7.2.2008, Termín 05
Příklad 01

V zadání bylo dáno nekonečně mnoho jazyků (jestli měl někdo pocit, že jenom čtyři, tak měl pocit špatný), o kterých víme, že jsou CF a nejsou regulární. Úkolem bylo najít jiný jazyk nad abecedou {a, b}, který má tuto vlastnost a dokázat to.

Hodnocení:
Najít jazyk, který splňuje řečenou vlastnost - 10 bodů
Typické srážky:
- jazyk, který je nad jinou abecedou (neumíte číst?): -7, pokud však byl jenom obměnou některého z jazyků ze zadání: -10
- jazyk regulární nebo neCF: -10 (ještě stále pak bylo možné dostat body za jednu ze dvou následuících částí, ale byl-li jazyk dokonce konečný, nedával jsem za celý příklad vůbec žádný bod.)
- formální chyby v zápisu jazyka: po -1
- nesmyslný dodatek "pro každé k \in N" (evidentně nemáte tuchu, co to znamená): rovněž -1

Dokázat, že je daný jazyk bezkontextový (napsáním CFG nebo PDA) - 15 bodů
Srážky za formální nedostatky - bod až dva za každý, generovala-li daná gramatika něco jiného: -10 až -15 bodů, podle závažnosti; jen náznak PDA místo korektní definice: -10 bodů, různé jiné specifikcé chyby (vždy je vysvětleno na papíře).

Dokázat, že daný jazyk není regulární - 20 bodů
Specifické nedostatky vždy vysvětleny na papíře.

Důkaz pomocí PL by měl vypadat nějak podle šabony:
Mějme libovolné n. Zvolíme si slovo w, |w| >= n, takto:
 w = ...
Všechna možná rozdělení slova w = xyz, |xy| <= n, y neprázdné, vypadají takto:
 x = ...
 y = ...
 z = ...
...
[Případně víc "skupinek". Pro každou skupinku pak:]
Zvolíme pumpovací konstantu i \in N0 takto:
 i = ...
Nyní
 x y^i z = ... = ... \not\in L   ==>   L není regulární.

Důkaz "bez omáčky" vypadá například nějak takto:
w = ...
x = ...
y = ...
z = ...
i = ...
x y^i z = ... = ... \not\in L

Podzim 2007, 7.2.2008, Termín 05
Příklad 02

???

Podzim 2007, 7.2.2008, Termín 05
Příklad 03

Ukolem bylo:
(a) Definovat, kdy je symbol X bezkontextove gramatiky G = (N, \Sigma, P, S) _nepouzitelny_.
(b) Definovat, kdy je bezkontextova gramatika _redukovana_.
(c) Odstranit nepouzitelne symboly z gramatiky:
G = ({S, A, B, C, D, E, F}, {a, b}, P, S)

P = {
S -> Baa | B
A -> aaaaa | Ca | Ea | Ba
B -> aaaaA
C -> DaEB | aAaaa | aaaS
D -> aDB | aAa | aCF | Sa | aBaaa
E -> aSaaE | aEbSa
F -> aaa | aaaaE | BaSa | Faaa
}

Reseni:
ad a) Symbol X z (N sjednoceno \Sigma) je nepouzitelny pokud v G neexistuje derivace tvaru S =>^* wXy =>^ wxy, pro nejaka w, x, y ze \Sigma^*
ad b) Gramatika je redukovana, pokud neobsahuje zadne nepouzitelne symboly.
ad c) Vysledna gramatika:
G = ({S, A, B, C}, {a}, P, S)

P = {
S -> Baa | B
A -> aaaaa | Ca | Ba
B -> aaaaA
C -> aAaaa | aaaS
}

Bodovani:
(a) (max. 7 bodu):
Za spatnou notaci jsem strhaval 1 az 2 body podle zavaznosti
Pokud jste definovali nepouzitelnost pouze jednoho typu, dostali jste 4 body.
(b) (max. 5 bodu):
Pokud jste na "redukovanost" gramatiky kladli prilis silne pozadavky (bez jendoduchyuch pravidel, necyklicka), strhaval jsem 2-3 body. Za to, ze jste napsali, ze neobsahuje epsilon pravidla jsem body nestrhaval, i kdyz jsem to preskrtnul, protoze je to zbytecne.
(c) (max. 25 bodu)
pokud jste napsali pouze algoritmus, daval jsem 5 bodu.
Za odstraneni nepouzitelnych symbolu pouze jednoho typu jsem daval 13 bodu a je jedno, jestli jste algoritmy pro odstraneni nenormovanych neterminalu a pro odstraneni nedosazitelnych symbolu provedli ve spatnem poradi nebo jste ten druhy z nich neprovedli vubec.

Podzim 2007, 7.2.2008, Termín 05
Příklad 04

v příkladě 4 jste měli zformulovat algoritmus pro syntaktickou analýzu obecných bezkontextových jazyků, uvést vstup a způsob určení výstupu.

Algoritmem je C-Y-K, popis najdete kupříkladu na slajdech doktora Strejčka nebo na Wikipedii. Vstupem je gramatika v CNF a testované slovo. Výstup se určí podle toho, jestli ve spočítané množině neterminálů, z nichž lze dané slovo odvodit, je iniciální neterminál, nebo ne.

Upozorňuji, že nikdo nechtěl nutně zadání algoritmu pseudokódem. Spousta lidí popsalo algoritmus větami a dostali za to 40 bodů.

Nejčastější prohřešky byly:
** Mlžení postupu získávání informací. To, jak se napočítají řádky tabulky z dosud spočítaných, je důležité. Za zmatek v tomto jsem často byl nucen strhnout i více jak 10 bodů.*
** Překlepy v indexech, drobné chyby v popisu algoritmu. Strhával jsem velmi málo, zvlášť pokud bylo poznat, že řešitel algoritmu rozumí, a jen se upsal.*
** Popis algoritmu konkrétním příkladem bez obecného postupu. Nic proti ilustracím výpočtu, ale samy o sobě nestačí. Za chybějící obecný postup jsem strhával až polovinu bodů.*
** Nepřesně zadaný vstup: někteří vynechali na vstupu gramatiku, jiní jen zapomněl, že má být v CNF. Bodové ztráty odpovídající, většinou do 5 bodů.*
** Neuvedení výstupu. Často jediná vada na kráse, chyba z nepozornosti. V takovém případě jsem strhával 2 body. Pokud bylo v řešení špatných více věcí, mohl se počet stržených bodů lišit.*
** Místo algoritmu C-Y-K se v některých řešeních objevilo pojednání o syntaktické analýze, LL analyzátorech apod. To není odpověď na otázku ze zadání, oceněno nejvýše 3 body (pokud dávalo aspoň trochu smysl).*

List 4 už se brzy naskenuje, ovšem z technických důvodů jej v ISu uvidíte až koncem týdne. Jelikož už to byl poslední termín, nemělo by to být výraznou překážkou ke splnění studijních povinností.

Podzim 2007, 7.2.2008, Termín 05
Příklad 05

v prikladu 5 jste meli dokazat nebo vyvratit nasledujici implikace:

a) bezkontextova gramatika G generuje regularni jazyk => G nema vlastnost sebevlozeni

b) L je neprazdny regularni jazyk a R neni regularni => L.R neni regularni

a)
implikace neplati
to muzeme ukazat napriklad timto protiprikladem:
Gramatika G={S,{a},P,S} kde P = {S->aSa|a|e} generuje regularni jazyk {a}* a zaroven ma vlastnost sebevlozeni.

Bodovani:
maximalni pocet bodu je 16
pouze odpoved, ze implikace neplati, je ohodnocena 4 body
pokud jste neuvedli, jaky jazyk Vase gramatika generuje, dostali jste 14 bodu

b)
implikace neplati
to muzeme ukazat napriklad timto protiprikladem:
L = Sigma*
R = {a^nb^n | n>= 0}
Pak L.R = Sigma*, coz je regularni jazyk

Bodovani:
maximalni pocet bodu je 16
pouze odpoved, ze implikace neplati, je ohodnocena 4 body
casta chyba byla, ze jste za R volili jazyk {a^nb^n | n > 0}. Pak ale L.R = Sigma.a^nb^n, coz neni regularni jazyk (zkuste si to dokazat pomoci P.L.). Za toto jste obdrzeli 10 bodu.*

Podzim 2007, 7.2.2008, Termín 05
Příklad 06

v prikladu 6 jste meli definovat Turinguv stroj (vcetne podminky na jeho prechodovou funkci)

Bodovani bylo nasledujici:
Uvedeni korektni devitice 4 body
Q - konecna mnozina stavu 2 body
Sigma - konecna mnozina reprezentujici vstupni abecedu 2 body
Gamma - konecna mnozina reprezentujici pracovni abecedu + Sigma je podmnozina Gamma 3 body
symbol pro levou koncovou znacku + nalezitost do mnoziny Gamma\Sigma 2 body
symbol pro prazdne policko + nalezitost do mnoziny Gamma\Sigma 2 body
Delta - totalni prechodova funkce 3 body
Spravny typ funkce Delta 10 bodu
pocatecni stav, akceptujici stav, zamitajici stav + nalezitost techto stavu do mnoziny Q 3 body
podminka na funkci Delta 3 body

Caste chyby byly neuvedeni konecnosti jednotlivych mnozina a neuvedeni prislusnosti jednotlivych symbolu do danych mnozin. Toto bylo penalizovano po 1 bodu za kazdou neuvedenou skutecnost. Hodne z Vas melo take problemy s typem funkce Delta.

Podzim 2008 , 7.1.2009, Termín 01
Příklad 01

V prvním příkladu bylo třeba rozeznat, který z jazyků je bezkontextový a který není, a své tvrzení dokázat. První jazyk bezkontextový nebyl, druhý ano. Protože důkaz, že jazyk není bezkontextový, je složitější, bylo za první část možno získat 30 bodů a za druhou část 20 bodů.
Zhruba třetinu bodů z každé části (10, resp. 7) jste dostávali za správné určení, že jazyk je/není bezkontextový. Zbytek bodů pak byl za důkaz, tj. Pumping lemma v prvním případě a gramatiku/zásobníkový automat v případě druhém.
Nejtypičtější chybou v první části bylo zapomenutí na některá rozdělení, za to se podle počtu a přesnosti popisu strhávalo 1 až 10 bodů. V případě, že bylo zjevné, že autor nechápe princip dokazování - úplně nevhodné slovo, špatně rozdělené, ... - bylo možné nedostat za pokus o PL žádný bod.
V druhé části jste body dostali, pokud byla z automatu/gramatiky vidět správná myšlenka, body se strhávaly za nedomyšlená řešení (některá slova neprošla, chyběly některé přechody) a za formální nedostatky v zápisu (max. 2 body).

L = {wcw', w,w' \in {a,b}^*, počet_a(w) = počet_a(w') A počet_b(w) = počet_b(w')}

Slovo pro pumpování zvolím a^n.b^n.c.a^n.b^n, jistě patří do jazyka pro každé n a má dostatečnou délku

Teď musím uvažovat všechna možná rozdělení na uvwxy, kde |vwx|<=n, v.x!=epsilon.
Stačí soustředit se na to, jakých různých hodnot může nabývat vwx; tady se to dalo rozepisovat různě složitě, ale stačí následující:

1) vwx = a^ib^j, i+j>0; pak i vx = a^ib^j, i+j>0, při volbě pumpovací konstanty 0 se v první nebo druhé části slova (zde "nebo" ve smyslu vylučovacím, celé vwx je buď před 'c', nebo za 'c') sníží počet a nebo b (zde "nebo" není ve smyslu vylučovacím), v druhé části se nezmění nic, takže se poruší rovnost počtu u alespoň jednoho znaku a slovo nepatří do jazyka

2) vwx = b^i.c.a^j, i+j>=0; pak možnosti
  2a) v nebo x obsahuje znak c, při pumpování s i=0 znak c zmizí, slovo nepatří do jazyka
  2b) vx = b^i.a^j, i+j>0; pak napumpování s i=0 sníží počet b v první polovině slova, ale v druhé ne, nebo (nevylučovací) sníží počet a v druhé polovině a v první ne; v každém případě se poruší rovnost, slovo nepatří do jazyka

Doufám, že se mi to podařilo napsat aspoň trochu srozumitelně... :)

osobne mi to pride robit lahsie, menej narocne a kratsie gramatiku, cize by som skor spravil to:

S -> A | B
A -> aAa | bA | Ab | c
B -> bBb | aB | Ba | c

kde sa nazaciatku rozhodnes coho bude v tom slove rovnako ci a alebo b a potom uz
len generujes pricom o pocet druheho terminalu sa uz nemusis starat. K tomuto potom mozes zostrojit automat PDA. Ja by som robil nieco na styl ze to bolo akceptovanie prazdnym zasobnikom a potom by som si to zase rozdelil na dva pripady rovnost poctu a, rovnost poctu b. Ak by bol prvy pripad pridaval by som na zasobnik a keby prislo b tak by som ho ignoroval, keby prislo c tak by som sa prepol do stavu uberania a. Podobne pre druhy pripad.

Podzim 2008 , 7.1.2009, Termín 01
Příklad 02

Priklad cislo 2 bol na konstrukciu regularnych vyrazov pre dane jazyky:
(a) L_1 = {w z {a,b,c}^* takych, ze w obsahuje zaroven podslova baa a cbc}
(16 bodov)
-spravne riesenie: az na male obmeny nieco tvaru
(a+b+c)^*.(baa.(a+b+c)^*.cbc + cbc.(a+b+c)^*.baa).(a+b+c)^*

(b) L_2 = {a,b}^* \ ({a}^*.{ba}.(b)^*)
(16 bodov)
-spravne riesenie: opat az na mensie obmeny podla sposobu uprav
a^* + a^*b + a^*ba.b^*a(a+b)^* + a^*bb(a+b)^*

Hodnotenie:
(a) Tuto cast mala valna vacsina dobre, najcastejsou chybou bol paradoxne zapis, kedy vyraz nebol regularnym vyrazom - obsahoval mnozinove zatvorky, zjednotenie a podobne. Pokial vsak bol popisany jazyk spravny, strhavala som za tuto chybu 1, maximalne 2 body. Niektory z vas napisali len jeden z danych scitancov, za co dostali 10 bodov.
V pripade, ze ste na to sli zo strany automatu a podarilo sa vam ho spravne skonstruovat, no jazyk ste uz nepopisali, mate 9 bodov.
Ak ste napisali vyraz alebo automat pre jazyk obsahujuci cbc alebo baa, dostali ste 5 bodov.

(b) Tato cast bola o dost zaujimavejsia a bohuzial aj menej uspesna. Idealne riesenie bolo skonstruovat automat pre a^*ba.b^* a, zamenit koncove za nekoncove a previest na regularny vyraz. Casto ste vsak zabudali doplnit prechodovu funkciu na uplnu, cim ste stratili velku cast riesenia. V pripade, ze ste ale mali dobry postup, dostali ste 8 bodov. Ak ste zabudli len na doplnenom "pekle" cyklit pod a, b alebo vam pri uprave uslo nejake epsilon, mate 14 bodov. V pripade zleho automatu, ale dobreho postupu to bolo 6 bodov.
Druha moznost bola "nejak to vymysliet". Tu ste vacsinou prisli aspon na a^*b^* co je vsak len +- tretina riesenia, 6 bodov. Potom podla toho kam az ste myslienky doviedli nejake body naviac.

Samozrejme sa nejake bodiky pridavali a odoberali v zavislosti na kombinacii chyb. Vase body mozete ocakavat zajtra, po naskenovani listov.

Podzim 2008 , 7.1.2009, Termín 01
Příklad 03

příklad 3 dopadl velmi dobře, většina z Vás jej zvládla zcela správně nebo jen s malými formálními nedostatky se srážkou maximálně 3 bodů z 35. Zde je několik připomínek k řešení a hodnocení:

(a) ZA pro analýzu shora dolů (20 bodů)

2 body za sedmici
1 bod za způsob akceptování
17 bodů za vlastní přechodovou funkci

Nejčastějšími problémy byly drobné formální nedostatky, za které bylo strháváno po bodu za každý typ (u vážnějších problémů i dva nebo tři body). Nejvýše jste za formální nedostatky ale mohli obdržet srážky 3 bodů.
Typicky: Vynechání závorek nebo záměna kulatých a složených závorek. Je rozdíl mezi množinou a uspořádanou n-ticí. Záměna = a -> u přechodové funkce.

Další frekventovanou chybou bylo sestrojení automatu, který akceptoval v koncovém stavu r. S tím souvisí i zavedení přechodu delta(q,eps,eps)={(r,eps)}, který měl být zřejmě použit při vyprázdnění zásobníku. Tento přechod může být ale realizován naprosto kdykoliv, proto automat akceptuje více než má. V lepším případě jste zavedli nové dno zásobníku.

Vážnější problém, který se vyskytoval poměrně často byla špatná definice delta(q,a,a)={(q,eps)} (a tudíž také delta(q,b,b)). -5 b.

V několika málo případech jste prováděli místo analýzy shora dolů analýzu zdola nahoru. V takovém případě jste mohli získat maximálně třetinu bodů za příklad.

(b) akceptující výpočet A nad slovem (15 bodů)

Nejčastějšími problémy byly opět drobné formální nedostatky, hodnocení viz výše.

Slovo čteme od začátku, ne od konce. Nemáme frontu, ale zásobník. To je zásadní neporozumění tomu, jak ZA pracuje. (max. 3 b.).

Někteří z Vás sestrojili správně automat pro analýzu shora dolů, ale výpočet prováděli jako by měli automat pro analýzu zdola nahoru. (max. 3 b.)

Zpracováváme pouze to, co je na vrcholu zásobníku. I když máme v zásobníku S které se bude přepisovat na epsilon, musíme počkat až to S bude na vrcholu a teprve pak jej na to epsilon přepsat.

Uvedené hodnocení jsou pouze orientační, Vaše hodnocení vzniklo na základě kombinace chyb, kterých jste se dopustili.

Podzim 2008 , 7.1.2009, Termín 01
Příklad 04

dnes vecer jsem doopravoval 4. priklad a zitra byste jej meli mit v ISu spolu s vysledky. V tomto prikladu bylo vasim ukolem **zformulovat algoritmus, ktery pro danou CFG bez epsilon pravidel zkonstruuje ekvivalentni gramatiku bez jednoduchych a epsilon pravidel**. Spravne reseni zde rozepisovat nebudu, jedna se o prislusny algoritmus z prednasek, pripadne ze skript. Pro dosazeni plneho poctu bodu nebylo nutne napsat presne tento algoritmus, ani pouzit pseudokodu. Bohuzel slovni popis byl v drtive vetsine pripadu (s jednou cestnou vyjimkou) velmi vagni a na rade mist nepresny. Na druhou stranu se vyskytlo nemalo reseni, ktera se hlavne v druhe casti od puvodniho algoritmu lisila a obdrzela mezi 35 a 40 body.

Pokud jste se vydali cestou podobneho algoritmu jako byl prezentovan na prednasce, tak bylo hodnoceni nasledujici (maximum 40b):

Popsani vysledne gramatiky a formalni zapis algoritmu: 5b
- zde jsem body az na vyjimky nestrhaval, koho se to tyka, ma v listu zduvodneni

Napocitani mnozin N_A: 15b
- v pripade, ze byla popsana pouze definice mnozin a nikoliv, jak je spocitat, pripadne vypocet byl _zcela_ chybne -10b
- pokud byl napsan pseudokod nebo popis postupu, tak za drobne formalni a technicke chyby podle charakteru -1b az -3b za kazdou, max. -7b

Napocitani mnoziny pravidel P': 20b
- pokud byl napsan pseudokod nebo popis postupu, tak za drobne chyby podle charakteru -1b az -3b za kazdou, max. -6b
- nezarazeni podminky, ze pouzivame jen pravidla, ktera nejsou jednoducha -5b nebo pokud jste vubec nesmerovali k odstraneni jednoduchych pravidel -10b

Pripady, kdy jste se snazili vymyslet vlastni algoritmus, nebo jste se prilis vzdalili od puvodni struktury, byly bohuzel vsechny neuspesne. Tyto postupy vedly k nekorektnim algoritmum, ktere bud pocitaly korektne jen pro nektere vstupy, nevracely gramatiku bez jednoduchych pravidel, nebo dokonce na nekterych vstupech cyklily. Takove algoritmy jsem hodnotil individualne a za alespon trochu smysluplny algoritmus, ktery trpel vyse zminenymi neduhy, jste mohli dostat az 10b. Body jsem odecital za nesrozumitelny popis a formalni a fakticke chyby. Zde bych chtel jen dodat, ze nepochybuji o tom, ze vetsina z vas by byla nejakym zpusobem schopna pro konkretni priklad nalezt vysledek, ale to zde nebylo ucelem.

Nejcastejsi chyby:
- nesrozumitelnost textu i pseudokodu
- nekteri z vas popsali i algoritmus pro odstraneni epsilon pravidel -- tyto algoritmy jsem pri opravovani ignoroval, priste prosim ctete zadani pozorne
- komentare u pseudokodu neodpovidaly tomu, co algoritmy opravdu delaly
- popsani toho, co chceme spocitat, bez toho, jak to udelame
- napocitani mnozin N_A a nasledne nevyuziti
- michani mnozin a jejich prvku, tj. vyrazy typu P' := A -> alfa misto P' := P' U {A -> alfa}
- terminologicke nepresnosti, napr. pravidlem je cele A -> alfa a nikoliv jen alfa (to je pravou stranou pravidla)
- demonstrovani algoritmu na konkretnim prikladu misto jeho popsani
- nekteri studenti si stale pletou slozene zavorky, tj. "{ }", ktere se pouzivaji pro oznaceni mnozin, a zavorky kulate, tj. "( )", ktere se pouzivaji u prvku relaci (n-tic)

Nakonec musim rict, ze jsem byl potesen tim, ze asi tretina z vas (33 studentu) dostala alespon 30 bodu. Tito studenti ve vetsine pripadu popsali vlastni verzi puvodniho algoritmu, coz vypovida o tom, ze algoritmus pochopili a nikoliv jen naucili (coz hodnotim velice kladne).

Podzim 2008 , 7.1.2009, Termín 01
Příklad 05

Regulární gramatika generující konečný bezkontextový jazyk existuje.
Například gramatika ({S},abeceda,{},S), tedy gramatika bez pravidel.

Vlastní gramatika s jednoduchými pravidly existuje.
Třeba gramatika({A,B,C},{x},{A->B, B->C, C->x},A).

Body:
Za špatnou odpověď až na výjimky 0. Mezi výjimky patří uvedení správné definice vlastní gramatiky (2 body).
Za správnou odpověď bez příkladu nebo aspoň vysvětlení 5 bodů.

Časté chyby:
** Gramatika v první části nebyla regulární.*
** Uvedená gramatika sice regulární, ale negenerovala konečný jazyk.*
** Podlehnutí mýtu, že regulární jazyky nejsou bezkontextové. To není pravda, jsou.*
** Zahrnutí zákazu jednoduchých pravidel do definice vlastní gramatiky. Vlastní gramatika může mít jednoduchá pravidla, ale nesmí v nich nastat cyklus umožňující A=>+A. Jiná věc je, že na vstupu algoritmu pro CNF musí být vlastní gramatika bez jednoduchých pravidel.*

Podzim 2008 , 7.1.2009, Termín 01
Příklad 06

Snažila jsem se hodnotit spíše podle toho, jestli definicím rozumíte, je to ale příklad na teorii, a proto se v hodnocení odráží i formální nedostatky (i když minimálně).

(a) Definujte typ funkce delta. 8b.

Správná odpověď: Q \ {q_acc,q_rej} x \Gamma -> Q x \Gamma x {L,R}

Hodnocení: 4 body za část před -> a 4 body za část za.

Nejčastější chyby:
záměna složených a hranatých závorek -1b.
Q x \Gamma -> Q x \Gamma x {L,R} 7b.
Q \ {q_acc,q_rej} x \Sigma \Gamma -> Q x \Gamma x {L,R} 6b.
Q \ {q_acc,q_rej} x \Gamma -> Q x \Gamma x {L,R} 6b.*

(b) Definujte konfiguraci Turingova stroje. 8b.

Správná odpověď: (q,z,n) \in Q x {y\sqcup^\omega|y \in Gamma*} x N_0, kde q je stav, z obsah pásky a n pozice hlavy na pásce.
\sqcup^\omega značí nekonečný řetězec znaků pro prázdné pole

Hodnocení: 2b. za uspořádanou trojici, 1b. za stav, 3b. za plně definovaný obsah pásky a 2b. za pozici.

(c) Definujte pojmy počáteční, akceptující a zamítající konfigurace.

Správné odpovědi:
počáteční (q_0, \triangleright w \sqcup^\omega, 0) 1b. za stav, 2b. za obsah pásky, 1 b. za pozici. \triangleright je levá koncová značka, \sqcup^\omega značí nekonečný řetězec znaků pro prázdné pole

akceptující (q_acc, z, n), 2b. za stav, po bodu za z, n
zamítající (q_rej, z, n), 2b. za stav, po bodu za z, n

Nejčastější chybou byla vágnost definicí -- akceptující konfigurace je když stroj přejde do q_acc, apod. V takovém případě jste dostali typicky 3b. ze 4.

(d) Jak se nazývá třída jazyků akceptovaných TS? 5b.
Správná odpověď: rekurzivně spočetné (typ 0, frázové)

1 bod jste dostali, pokud jste odpověděli rekurzivní.

Podzim 2008 , 15.1.2009, Termín 02
Příklad 01

v příkladu 1 bylo Vaším úkolem navrhnout zásobníkový automat akceptující jazyk c^n.w, kde w \in {a,b}^* a n+počet znaků a = počet znaků b.

Většina z Vás příklad řešila tak, ze jste zadefinovali automat a vypsali jednotlivé položky přechodové funkce. Někteří zkusili navrhnout gramatiku pro jazyk a pak odvodit automat pro syntaktickou analýzu. Toto řešení je myšlenkově taky v pořádku (že někteří použili rozšířený ZA jsem tolerovala, ač to v zadání nebylo uvedeno), ale nikomu se nepodařilo správně navrhnout gramatiku. Ti, kdo navrhovali automat, ale popsali jej neformálně (textem či obrázkem), mohli dostat max. 30 bodů - z takového popisu není zřejmé, zda jste skutečně domysleli všechny varianty.

Při návrhu automatu bylo potřeba podchytit dvě základní vlastnosti slov z jazyka:
a) slova jsou tvaru c^n.w, tedy znaky c mohou být jen na začátku slova
b) počet znaků c + počet znaků a = počet znaků b
Část a) je výrazně jednodušší, proto byla hodnocena 10 body, zbytek 30. Při počítání znaků bylo nejčastější chybou, že jste na zásobník ukládali znak X za každé a,c a při výskytu b jste jeden znak X odmazali. To ale funguje jen pro ta slova, kdy každý prefix slova obsahuje nejvýše tolik znaků b, kolik obsahuje znaků a,c. Všichni, kdo tuto variantu nedomysleli, mají v písemce napsáno něco typu "automat neakceptuje slovo ccbbba" a o 15 bodů míň. Správné řešení muselo uvažovat variantu, že příjde znak b, když zásobník bude prázdný (nebude co mazat), pak se tento znak musí na zásobník uložit (buď jako jiný symbol zásobníkové abecedy, nebo v jiném stavu) a zase bude smazán příchodem znaku a. Další srážky byly za smazání dna hned na začátku při akceptování prázdným zásobníkem - pak automat nemůže akceptovat slovo abab, protože už po ab vymaže zásobník a pak už se přechodová funkce "nemá čeho chytit". Toto jsem hodnotila -5 body, stejná srážka byla za opomenutí případu, kdy n=0, tj. slovo neobsahuje znaky c.

Konečně nejvýše 3 body bylo možné ztratit za formální nedostatky zápisu automatu jako sedmice a přechodové funkce - nemálo řešení si ji pletlo s přepisovacími pravidly.

Podzim 2008 , 15.1.2009, Termín 02
Příklad 02

Do ISu už by měly být skenovány vaše příklady na M-N větu z druhé zkoušky.

Zadání: Dokažte M-N větou, že L={a^i.b^j | j nerovno 2i} není regulární.

Řešení: Díky M-N větě stačí dokázat, že prefixová ekvivalence ~L má nekonečně mnoho tříd. To zjevně platí, pokud dokážeme, že nekonečná množina slov {a}* neobsahuje žádnou dvojici slov, která by byla v relaci ~L. Sporem: nechť a^i ~L a^j pro i nerovno j. Pak přiřetězením slova b^(2i) dostaneme jednou slovo a^i.b^(2i), které do L nepatří, podruhé a^j.b^(2i), které do L patří, neboť 2j není rovno 2i díky předpokladům a injektivitě násobení dvojkou. To ale značí, že a^i nebylo v relaci ~L s a^j, což je spor.

Časté chyby:
Mnozí z vás dokazovali o konkrétní dvojici slov, že nejsou v relaci ~L, to je ale málo -- musíme najít nekonečně mnoho slov, tak že žádná dvě z nich nejsou ve ~L.
Jiní psali, že L není regulární, protože není akceptován konečným automatem (bez důkazu), a proto index ~L není konečný. Význam M-N věty je mimojiné v tom, že umožňuje podat důkaz neexistence konečného automatu pro daný jazyk pomocí nekonečnosti ~L.
Maličkost, která není sama o sobě chybou, ale svědčí o neúplném pochopení, je, že mnozí z vás nejdříve převedli problém regularity L na regularitu jeho doplňku, a pro ten se snažili použít M-N větu. Uvědomte si, že prefixová ekvivalence jazyka L i jeho doplňku je stále tatáž relace!
Nejsmutnější řešení byla ale ta, kde řešitel odevzdal papír chaoticky popsaný písmeny u, v, w, a, b, i, j a znakem ~, občas doplněný útržky vět. Bohužel takový řešitel nebyl jeden. Ačkoli jinak za náznak toho, že víte, co je M-N věta, nebo jak by probíhal důkaz, byly aspoň nějaké body, za výše popsané "řešení" bylo 0 bodů -- základem matematické kultury je smysluplné vyjadřování. Nevím, jak by se dotyčným líbilo, kdyby takto vypadala skripta.

Podzim 2008 , 15.1.2009, Termín 02
Příklad 03

Nejčastější chybou bylo nenapsání těch dvou uspořádání neterminálů, o které jsme vás v zadání žádali. Za to jste utržili od dvou do pěti záporných bodů, podle toho, jak moc se dalo uspořádání vyčíst z obrázků/postupu.

Celý proces odstranění levé rekurze byl hodnocen 25 body, 15 za přímou levou rekurzi a 10 za zbytek. Za převod nelevorekurzivní gramatiky do GNF jste mohli dostat až 15 bodů, chybějící čtveřice znamenala ztrátu jednoho bodu a porušení pravidla o neterminálech za terminálem na pravé straně až 5.

Pokud se vám nepodařilo nic než konstatovat, že gramatika je vlastní a tedy vhodná pro odstraňování levé rekurze, dostali jste 3 body.

Podzim 2008 , 15.1.2009, Termín 02
Příklad 04

hodnocení příkladu 4 brzo uvidíte ve svých poznámkových blocích. Jednalo se o **zformulování algoritmu pro deterministickou syntaktickou analýzu**. Měli jste také uvést vstup a způsob, jak se určí výstup algoritmu. Správné řešení je např. algoritmus CYK. najdete ve slidech z přednášky 8.12. Příklad byl celkem za 40 bodů a hodnocení bylo následující.

- 5 b. za vstup (1 za bezkontextovou gramatiku, 3 za CNF, 1 za slovo). Velmi často jste tady zapomínali, že gramatika musí být v Chomského normální formě.

- 10 b. za napočítání množin T_i,1

- 20 b. za napočítání zbylých množin T_i,j

- 5 b. za určení výstupu (tedy pokud S \in T_1,n, pak "ano" jinak "ne"). Pokud jste jen uvedli, co je výstupem a chybělo jak jej získat, obdrželi jste 2 b.

Za formální nedostatky mohly být strženy nějaké body, maximálně ale 3.

Nejčastějším problémem byla vágnost, nepřesnost a neúplnost algoritmu. Nejlepší variantou je pseudokód podobný jako ve slidech. Slovní popis obvykle vnáší do algoritmu nejasnosti. Důraz byl ale kladen na to, zda jste algoritmu porozuměli.

Dalším problémem bylo, že místo deterministické analýzy jste prováděli nedeterministickou shora dolů nebo zdola nahoru (max. 10 bodů).

Ke každému zásobníkovému automatu nemusí existovat ekvivalentní deterministický zásobníkový automat.

Podzim 2008 , 15.1.2009, Termín 02
Příklad 05

list obsahoval dva příklady. V obou se po Vás chtělo rozhodnout o existenci případně neexistenci bezkontextové gramatiky jistých vlastností a Vaše tvrzení potom dokázat. Správná odpověď v obou případech byla, že gramatika existuje.

a) CFG generující neprázdný jazyk, která je zároveň v CNF i GNF:

Např. G = ({S}, {a}, {S -> a}, S)

b) CFG, která je zároveň necyklická a nejednoznačná:

Např. G = ({S}, {a}, {S -> SS | a}, S), slovo "aaa" potom má dva různé derivační stromy odpovídající těmto levým derivacím:
1) S => SS => aS => aSS => aaS => aaa
2) S => SS => SSS => aSS => aaS => aaa

Bodování:

Platí pro a) i b):

(max 16b) Za odpověď ANO byly 4 body, zbylých 12 bylo za správný příklad gramatiky. V několika případech jsem za příklad gramatiky udělil 8 bodů. Byly to případy, kde gramatika sice nebyla úplně správně, ale vypadalo to, že důvod je spíše v nepozornosti, než neznalosti.

Podzim 2008 , 15.1.2009, Termín 02
Příklad 06, 07

Příklad 6
V tomto příkladu se po Vás chtělo zadefinovat pojem gramatika. Věci, které tam měli zaznít jsou následující:
- je to uspořádaná čtveřice obsahující 3 množiny a jeden speciální prvek množiny první
- množiny jsou neprázdné a konečné
- množiny neterminálů (ta na prvním místě čtveřice) je disjunktní s množinou terminálů (druhá množina)
- popis pravidel - např. pravidla z množiny třetí jsou dvojice řetězců ze symbolů prvních 2 množin, přičemž první prvek obsahuje alespoň jeden symbol z množiny neterminálů
- S je prvek množiny neterminálů a je to iniciální neterminál

za popis čtveřice jsem dával 4 body, 4 za popsání množiny neterminálů, 3 za popsání množiny terminálů, 2 za napsání že množiny jsou disjunktní, 4 za popis množiny pravidel a 3 za popis iniciálního neterminálů

Příklad 7
za každou operaci 2 body

Podzim 2008 , 25.1.2009, Termín 03
Příklad 01

L1 = {wcw'|w,w'\in {a,b}*, #a(w) + #b(w) = #a(w') + #b(w') }
L2 = {wcw'|w,w'\in {a,b}*, #a(w) = #b(w) = #a(w') = #b(w') }

Ad L1:
Jazyk je bezkontextovy. Pro dukaz teto skutecnosti stacilo napsat odpovidajici gramatiku nebo zasobnikovy automat. Mohli jste ziskat az 20 bodu.
Priklad spravne gramatiky:
G=({S},{a,b,c},P,S)
P = {S -> aSa|aSb|bSb|bSa|c}

Caste chyby.
Formalni nedostatky jako napr. chybejici definice (hlavicka) gramatiky nebo automatu, chybejici zpusob akceptovani, nepresne definovana prechodova funkce a podobne nepresnosti. Za tyto chyby Vam bylo strzeno 0 az 3 body dle zavaznosti.
Dalsi chyby byly v nepresnem navrhu gramatiky ci automatu jako napr. Nedomysleni okrajovych situaci (c patri do L1 ale epsilon nepatri do L1), zapomenuti vyprazdneni zasobniku pri akceptovani prazdnym zasobnikem, opomenuti nejakeho prechodu a podobne. Za tyto chyby Vam bylo strzeno 3 az 5 bodu dle zavaznosti.
Pokud vase gramatika ci automat obsahoval nejakou zasadni koncepcni chybu (napr. jste akceptovali resp. generovali pouze jazyky kde w'= w^R a jine varianty nebo vas automat mohl akceptovat driv nez vyprazdnil zasobnik) ziskali jste maximalne 10 bodu.
Pokud jste odpovedeli pouze ano bez uvedeni automatu ci gramatiky dostali jste 4 body.

Ad L2:
Jazyk neni bezkontextovy. Pro dukaz teto skutecnosti jste mohli vyuzit P.L. pro bezkontextove jazyky. Mohli jste ziskat az 30 bodu.
Priklad spravneho pouziti P.L.
Necht n \in N je konstanta z P.L.
Volime slovo z = a^nb^nca^nb^n
Nyni uvazime vsechna rozdeleni slova z na podslova u,v,w,x,y takova, ze
|vx|>0 a |vwx|<= n a pro kazde toto rozdeleni ukazeme ze existuje i \in N_0
takove, ze slovo ux^iwy^iz nepatri do jazyka L2.
Uvazme rozdeleni kde podslovo vwx obsahuje pouze znak(y):

'a'- volime i = 0 a dostavame ze
ux^iwy^iz = a^{n-k}b^nca^nb^n nebo
ux^iwy^iz = a^nb^nca^{n-k}b^n
pro n>=k>0
'b'- volime i = 0 a dostavame ze
ux^iwy^iz = a^nb^{n-k}ca^nb^n nebo
ux^iwy^iz = a^nb^nca^nb^{n-k}
pro n>=k>0
'a,b'- volime i = 0 a dostavame ze
ux^iwy^iz = a^{n-k}b^{n-l}ca^nb^n nebo
ux^iwy^iz = a^nb^nca^{n-k}b^{n-l}
pro n>=l+k>0 a k,l>=0
'a,b,c'- volime i = 0 a dostavame ze
ux^iwy^iz = a^nb^{n-k}ca^{n-l}b^n
pro n>l+k>0 a k,l>=0 nebo
ux^iwy^iz = a^nb^{n-k}a^{n-l}b^n
pro n>l+k>=0 a k,l>=0

Jak je videt, pro kazde pripustne rozdeleni jsme ukazali, ze existuje i \in N_0,
takove ze slovo ux^iwy^iz nepatri do jazyka L2 a tudiz z P.L. pro bezkontextove
jazyky plyne, ze jazyl L2 neni bezkontextovy.
Caste chyby.
Nepochopeni P.L. nebo jeho spatne pouziti. Typicky jste nezvolili slovo nebo zvolene slovo bylo spatne (nesplnovalo podminky P.L. nebo pro nej existovalo "spatne" rozdeleni). V tomto pripade jste mohli ziskat maximalne 8 bodu.
Dalsi chybou bylo neuvazeni vsech platnych rozdeleni nebo jejich nepresne popsani:
Uvedeni jen jednoho konkretniho rozdeleni znamenal ziska maximalne 10 bodu. Uvedeni nekolika konkretnich rozdeleni znamenal ziska max 12 bodu. Pokud jste zapomeli na urcite druhy rozdeleni (viz spravne reseni) strhaval jsem 5 az 10 bodu.
Pokud vam chybeli jen nektera konkretni rozdeleni nebo jste vase rozdeleni nepopsali presne strhaval jsem 1 az 5 bodu.

V tomto prikladu bylo dulezite zvolit vhodny zpusob jak popsat vsechna pripustna rozdeleni zvoleneho slova z na podslova u,v,w,x,y a jak o nich ukazat, ze existuje i \in N_0, takove ze slovo ux^iwy^iz nepatri do jazyka L2.
Ruzne zvolene zpusoby pak implikovaly ruznou obtiznost potrebne argumentace. Nekteri z Vas zvolili zpusoby, ktere nebyly zcela presne a bylo jim strzeno 3 az 10 bodu dle zavaznosti. Pokud jste odpovedeli pouze ne bez dukazu dostali jste 4 body

Podzim 2008 , 25.1.2009, Termín 03
Příklad 02

Vaším úkolem bylo napsat regulární výraz E popisující jazyk akceptovaný následujícím konečným automatem A:

		a	b
->	1	3	2
<-	2	1	2
<-	3	2	1

V zásadě jste mohli postupovat dvěma způsoby. Doporučeným postupem řešení bylo použití algoritmu z přednášky, mohli jste se však rovněž rozhodnout pro alternativní metodu. V takovém případě se však po Vás chtěl důkaz rovnosti L(A)=L(E). Bodování do značné míry záviselo na způsobu řešení, pro který jste se rozhodli. Maximálně jste mohli získat 30 bodů.

1. Použití standardního algoritmu
Výsledný RE závisel na tom, v jakém pořadí jste odstraňovali původní stavy. Ideální bylo odstraňovat stavy v pořadí 3-2-1, v takovém případě totiž docházelo k nejmenšímu narůstání délky RE u jednotlivých hran, čímž se rovněž nižovala pravděpodobnost, že uděláte chybu.

Body jsem rozděloval následovně:
Správná inicializace algoritmu (přidání nového počátečního a koncového stavu a epsilon-přechodů): 3 body (ke správnému výsledku jste v jistém případě mohli dojít i bez přidání nového poč. stavu, v takovém případě jsem body nestrhával).
Za správné odstranění každého původního stavu: 8 bodů (celkem tedy 24 bodů). V případě, že jste při odstraňování stavu zapomněli na nějakou hranu, strhával jsem v příslušném kroku 2 až 8 bodů (podle počtu těchto chyb). Typicky jste například při odstranění stavu 3 (pakliže jste postupovali ve výše zmíněném pořadí) zapomněli přidat hranu z 1 do nového konc. stavu s návěštím 'a'.

Za zapsání výsledku: 3 body. Tohle byl spíš takový bonus, implicitně jsem bral jako výsledný ten RE, který byl návěštím poslední hrany reg. přech. grafu po ukončení algoritmu. Přesto bych Vás rád požádal (týká se to zejména kolegů, kteří podle algoritmu nepostupovali), abyste vždy řádně a jednoznačně vyznačili výsledek (nejlépe klasickým dvojitým podtrhnutím, kterému Vás učili kdysi dávno na ZŠ :)). Vyznat se v některých Vašich písemkách chtělo značnou dávku invence :).

Někdy Váš postup připomínal standardní algoritmus, ovšem v některých zásadních aspektech se lišil - typicky v tom, jakým způsobem zachováváte "kružnice" při odstraňování stavu (například 1 ->^a 2 ->^b 1 kde ->^a je hrana s návěštím 'a'). V takovém případě probíhalo bodování podle části 2.

2. Použití jiného postupu
Zde bylo 20 bodů za RE a 10 bodů za důkaz/zdůvodnění. Až na jeden případ vhodné zdůvodnění nenapsal nikdo (a tento případ byl bohužel jediným, kdy se podařilo dostat ke správnému výsledku bez použití algoritmu). Dával jsem však i nějaké body za dobrou myšlenku (např. počítání reg. výrazů pro jazyky všech slov převádějících automat do stavu 2, resp. 3).

Většina z Vás zvolila právě nějaký takový postup, tj. snažili jste se "ručně" napočítat zmíněné regulární výrazy. Prakticky ve všech případech Vám však ve výsledném jazyce chybělo nekonečně mnoho slov, typicky slova tvaru ab(ba)^nb, n≥1, či a^(3n+1), n≥1. Pokud jste si však dali pozor na to, aby Vámi zadaný výraz popisoval nějakou (nekonečnou) podmnožinu jazyka L(A), dostali jste 12 bodů (a to i v případě triviálních odpovědí typu bb*).

16 bodů dostal ten, pro jehož RE platilo L(A)⊆L(E), byl-li jazyk L(E) dostatečně netriviální. (Mám za to, že postihnout daným výrazem všechna slova jazyka L(A) je obtížnější, než postupně přidávat slova, která automat převedou do konc. stavu.) Takové případy však byly jen 2, u ostatních byl jazyk L(E) (aniž byste si to uvědomili) příliš triviální (např. Σ* nebo Σ*\{eps}). Takový jazyk nelze použít ani jako hrubou "aproximaci".

Pokud v jazyce popsaným Vaším RE chybělo nekonečně mnoho slov z jazyka L(A) a zároveň v něm přebývalo nekonečně mnoho slov z co-L(A), žádný bod jste bohuželnezískali.

Globální hodnocení:
Nejvýše dva body jsem strhával, pokud jste ve výsledku neuvedli RE (sjednotítka namísto +, použití pozitivní iterace apod.), ostatní formální nedostatky jsem nechával bez povšimnutí. Výsledný počet bodů pochopitelně závisí na kombinaci nalezených chyb.

Podzim 2008 , 25.1.2009, Termín 03
Příklad 03

Vašou úlohou bolo k danej gramatike skonštruovať rozšírený PDA pre nedeterministickú syntaktickú analýzu zdola nahor, rozhodnúť akým spôsobom akceptuje a popísať rozdiel v použitej notácii zapísaného rozšíreného PDA od tej štandardnej (spolu 22b) a následne uviesť akceptujúci výpočet automatu nad slovom 01120 (18b).

V prvej časti vám boli body rozdelené nasledovne:
3b za sedmicu
1b za uvedenie akým spôsobom akceptuje váš automat (toleroval som aj odpoveď prázdnym zásobníkom aj napriek tomu, že sa to líšilo od konštrukcie prezentovanej na prednáške v prípade, ak to korešpondovalo s vašim automatom)
2b za popísanie rozdielu v notácii zápisu rozšíreného PDA (vrchol zásobníka je vpravo) od štandardnej (vrchol je vľavo)
Zvyšných 16 bodov ste mohli získať za definíciu prechodového zobrazenia a medzi jeho konkrétne "pravidlá" boli podelené takto:
3b spolu za pravidlá na čítanie, teda pravidlá typu \delta(q,0,\eps) = {(q,0)}
9b spolu za redukčné pravidlá a špeciálne ešte 2b za pravidlo \delta(q,\eps,1) = {(q,A),(q,B)}
2b za pravidlo \delta(q,\eps,$S) = {(r,\eps)}, kde $ predstavujúce znak pre nové dno
Okrem toho som ešte strhával po bode za rôzne formálne nedostatky. Špeciálne o 2b ste prišli ak ste neuvádzali množinové zátvorky pri definovaní \delta.

V druhej časti patrili 3 body správnemu zápisu konfigurácie automatu: (stav, neprečítaná časť slova, aktuálny obsah zásobníka). Mnohí z vás prehadzovali poradie zložiek (-2b) alebo vôbec nepísali stav (-1b). Samozrejme, ak niekto zmienil, že stav bude všade až na konci rovnaký a preto ho nepíše, tak to bolo v poriadku.
Správny výpočet mal 13 krokov. V prípade, že ste neuviedli správny výpočet, niektoré kroky vám chýbali alebo váš výpočet bol neakceptujúci, tak ste body dostali v pomere vášho riešenia k tomu správnemu.
Ak vám síce výpočet chýbal (alebo bol nekompletný), ale uviedli ste aspoň správne odvodenie slova alebo jeho derivačný strom, tak som vám dal 2 alebo 3 body.
Za výpočty, kde ste načítavali slovo od konca, prípadne menili obsah zásobníka niekde uprostred a nie od vrchu, som sa snažil dať aspoň pár bodíkov za to, čo bolo správne.

Posledná poznámka sa týka prípadu, že ste robili analýzu zhora nadol. Vtedy ste mohli získať maximálne 1/3 bodov v oboch častiach.
Zaujímavé bolo, koľkým z vás výpočet vôbec nekorešpondoval s napísaným algoritmom. :-(A bežne sa stávalo, že automat síce robil analýzu zdola zahor, ale výpočet už predstavoval analýzu zhora nadol.

Podzim 2008 , 25.1.2009, Termín 03
Příklad 04

Vašim úkolem bylo **zformulovat algoritmus převodu NFA na DFA**, který zaručí, že výsledný DFA bude mít totální přechodovou funkci a nebude obsahovat nedosažitelné stavy. Spravne reseni zde rozepisovat nebudu, jedna se o prislusny algoritmus z prednasek, pripadne ze skript. Pro dosazeni plneho poctu bodu nebylo nutne napsat presne tento algoritmus, ani pouzit pseudokodu. Bohuzel slovni popis byl v drtive vetsine pripadu (az na 2 vyjímky) velmi vagni a na rade mist nepresny.

Pokud jste popisovaly odpřednášený algoritmus, tak za hrubé nedostatky (špatné podmínky v cyklech atd.) jsem strhával 3b, za drobnější prohřešky 2 a za opomenutí(množ. závorky, očárkování množin...) 1b.

Pokud jste popisovaly algoritmus stylem "Z kuchařky mojí babičky", dopouštěly jste se řady nepřesností. Často jste zapomínaly popsat, jak určit přechody pro množinu stavů. Používali jste formulace jako "podívám se, kam se dostanu" apod. Za popisy, ve kterých jsem našel alespoň základní myšlenku, jsem dával 10 bodů.

Nejčastější chyby:
- nesrozumitelnost (V jedné větě se např. vyskytlo nověvzniknutý, nový a vzniknutý stav pro 3 různé stavy)
- zaměňovali jste často automat za gramatiku
- nedotahovali jste myšlenky do konce
- odstranění nedosažitelných stavů původního automatu je sice věc pěkná, ale že byl stav dosažitelný v původním automatu neznamená, že ten samý stav bude dosažitelný v automatu novém
- psali jste algoritmy pro odstranění nepoužitelných stavů a pro ztotálnění přechodové funkce, které ale samy o sobě automat nedeterminizují (což byl hlavní cíl)

Podzim 2008 , 25.1.2009, Termín 03
Příklad 05

Zadání (a):
Rozhodněte, zda existuje gramatika, která není regulární, je bezkontextová a generuje regulární jazyk.

Řešení: Ano, existuje. Například gramatika G=({S},{a},{S->S},S) generuje prázdný jazyk, který je regulární. Přitom G není regulární, ale je bezkontextová.

Bodování: Za pouhou správnou odpověď byly nejvýše 3 body. Pokud byla odpověď doplněna o gramatiku, jak zadání požadovalo, bylo bodů až 15. Častou chybou bylo uvedení pravidel bez gramatiky (-1 bod), což není jednoznačná odpověď. Mezi další chyby patřilo uvedení regulární gramatiky, uvedení gramatiky, která negeneruje regulární jazyk, a další variace.

Zadání (b):
Rozhodněte, zda existuje cyklická bezkontextová gramatika bez jednoduchých pravidel.

Řešení: Úvodní poznámka: gramatika je cyklická, pokud obsahuje neterminál A takový, že je možné z A odvodit v nenulovém počtu kroků opět A, tj. A=>+A.
Možné odpovědi jsou ano i ne. Záleží na tom, zda pod pojmem bezkontextové gramatiky rozumíme gramatiku bez ε-pravidel, nebo ne.
Pokud povolíme ε-pravidla, odpověď je: ano, například
G=({S},{a},{S->SS,S->ε},S), kde máme S=>SS=>S.
Pokud ε-pravidla nepovolíme, odpověď je ne. Přímý důkaz: nechť je gramatika cyklická. Máme tedy neterminál A a odvození tvaru A=>α=>*A, kde A->α je pravidlo gramatiky. Přitom α není terminál, protože z terminálu nelze pokračovat v derivaci. Zároveň ale |α|=<1, neboť v gramatice nejsou zkracovací pravidla (ε-pravidla), která by umožnila vygenerovat z α zpět A. Tedy α je neterminál a A->α jednoduché pravidlo. Tedy každá cyklická gramatika bez ε-pravidel obsahuje jednoduchá pravidla.

Bodování: Vzhledem k nejednoznačnosti odpovědi byly body přiděleny pouze za zdůvodněné odpovědi. Častý problém byl, že řešitel neznal definici cyklické gramatiky. Byla-li jako důkaz odpovědi "ano" uvedena necyklická gramatika, byla tato odpověď ohodnocena 0 body a nejčastěji poznámkou "N.C." (není cyklická). Podobná chyba byla uvedení gramatiky s jednoduchými pravidly. V důkazech byl často problém se záměnou odvození A=>+A a pravidla A->A. Pokud řešitel uvedl důkaz odpovědi "ne", který nebyl úplný, ale obsahoval podstatnou část myšlenky, nějaké body dostal.

Podzim 2008 , 25.1.2009, Termín 03
Příklad 06

na tomto listu jste meli za ukol napsat definice 3 pojmu, tyto definice jsou uvedeny ve skriptech a ve slajdech. Za kazdou definici jste mohli ziskat az 10 bodu, celkem tedy 30. Hodnoceni bylo nasledujici:

a) Definujte, kdy ma CFG vlastnost sebevlozeni
CFG G = (N,\Sigma,P,S) ma vlastnost sebevlozeni, pokud existuji A \in N a u, v \in \Sigma^+ takova, ze A =>^+ uAv. ... 10b

- *zde hodne zalezelo na kontextu a celkovem stylu definice*
- *pokud jste napsali misto "A =>^+ uAv" nejaky vyraz s A na leve strane a uAv na prave, dostali jste alespon 3b*
- *pokud jste navic napsali neco ve smyslu, ze u, v patri do \Sigma^+, tak jste dostali 2b*
- *zbyle body jste dostali za ostatni nalezitosti, tj. pokud byly v poradku kvantifikatory, pokud jste napsali odkud se vzaly pouzite symboly A, u, v, atd.*

b) Definujte, kdy je relace ~ pravou kongruenci:

~ musi byt relace ekvivalence ... 3b

- zde jste bud body dostali, nebo ne

\forall u,v,w \in \Sigma^* : u ~ v => u.w ~ v.w ... 7b

- *alternativne lze pouzit a \in \Sigma misto w*
- *pokud jste pouzili spojku ekvivalence misto implikace (tj. <=> misto =>), tak -4b*
- *pokud jste pouzili existencni misto vseobecneho kvantifikatoru -3b*
- *pokud jste dostatecne nepopsali, co jsou u,v,w -1b nebo -2b*
- *pokud jste do definice nejakym zpusobem pridali nejaky jazyk L, tak -1b az -3b podle zavaznosti*
- *pokud jste napsali neco, co alespon vzdalene pripomina vyse uvedenou definici, tak jste dostali (pres vsechny uvedene penalizace) alespon 1b*

c) Definujte, kdy je CFL vnitrne viceznacny:

Jazyk L je vnitrne (inherentne) viceznacny, pokud kazda gramatika, ktera jej generuje, je viceznacna. CFG G je viceznacna prave kdyz existuje w \in L(G) majici alespon dva ruzne derivacni stromy. ... 10b

- *alternativne bylo mozne pouzit pojmy jako nejednoznacny a jednoznacny, pokud byly radne zadefinovany*
- *v pripade formalnich chyb jsem strhaval 1b az 3b, podle zavaznosti*
- *pokud jste uvedli pouze prvni cast, nelze to bohuzel uznat jako uplnou definici a dostali jste 6b*
- *pokud jste z definice uplne vynechali gramatiku, ale zminili jste, ze je treba, aby existovali alespon dva derivacni stromy, tak jste dostali 1b az 4b, podle toho, jak vase (v tomto okamziku jiz spatna) definice davala smysl*
- *v pripade, ze jste napsali neco ve smyslu "... existuje viceznacna gramatika, ktera jazyk generuje ..." a druhou cast jste meli dobre, dostali jste 2b*

Poznamky:
Nekolikrat jsem se setkal s tim, ze student zadefinoval dvakrat ruzne tentyz pojem (typicky u prave kongruence -- jednou slovne a podruhe pomoci symbolu). V techto pripadech se velmi casto stavalo, ze se obe dve definice vyrazne lisily, coz vypovida o tom, ze autor bud nerozumi definici, nebo neni schopen sve myslenky zformulovat -- v kazdem pripade to neni dobry zpusob jak upozornovat na svou neznalost.

Podzim 2008 , 4.2.2009, Termín 04
Příklad 01

Byly dve verze zadani:
Verze 1: Navrhnete gramatiku v GNF generujici jazyk L = {a^n.b^{n + m}.c^m | n, m >= 0}

Reseni mohlo vypadat napriklad takto:
G = ({S, X, Y, B, C}, {a, b, c}, P, S)
P = {
S -> aX | bY | aXBY | \epsilon
X -> aXB | b
Y -> bYC | c
B -> b
C -> c
}

Bodovani:
- Zhruba receno, 20 bodu bylo za to, ze gramatika generuje dany jazyk (prvni cast), a zbylych 20 bodu bylo za to, ze gramatika je v GNF (druha cast). Pokud ovsem mela gramatika zavaznejsi nedostatky, tak za to, ze je v GNF, plnych 20 bodu nebylo, ale byl z nich strzen zhruba stejny pocet bodu, jako z tech 20 bodu za prvni cast.
- Pokud gramatika nebyla v GNF, bylo strzeno 2-20 bodu, podle toho "jak moc nebyla" v GNF
- Pokud gramatika negenerovala nektere "krajni pripady" (napr. slova a^n.b^n) bylo strzeno 2-10 bodu.
- Pokud mela gramatika zavaznejsi nedostatky, bylo strzeno 10-20 bodu z prvni casti (a odpovidajici pocet bodu z druhe casti), podle toho, jak "vyznamnou" cast jazyka L negenerovala.
- Nekteri z vas si nezavedli pro generovani casti a^n.b^n novy neterminal a pouzili korenovy neterminal S, coz zpusobilo, ze gramatika generovala i slova, kde pismena c mohla byt pred pismeny b. Pokud byla gramatika jinak v poradku, bylo za to celkem 20 bodu.
- Za pouhou definici GNF byly celkem 2 body
- Za zkonstruovani automatu misto gramatiky bylo celkem 5 bodu
- Pokud gramatika nebyla v GNF a negenerovala alespon nejakou "vyznamnou" cast jazyka L, nebyly za to zadne body

Verze 2: Navrhnete zasobnikovy automat akceptujici jazyk L = {a^n.b^{n + m}.c^m | n, m > 0}

Reseni mohlo vypadat napriklad takto:
A = ({q_0, q_1, q_2, q_3, q_4}, {a, b, c}, {A, Z}, q_0, Z, {q_4})

\delta(q_0, a, Z) = {(q_0, AZ)}
\delta(q_0, a, A) = {(q_0, AA)}
\delta(q_0, b, A) = {(q_1, \epsilon)}
\delta(q_1, b, A) = {(q_1, \epsilon)}
\delta(q_1, b, Z) = {(q_2, AZ)}
\delta(q_2, b, A) = {(q_2, AA)}
\delta(q_2, c, A) = {(q_3, \epsilon)}
\delta(q_3, c, A) = {(q_3, \epsilon)}
\delta(q_3, \epsilon, Z) = {(q_4, \epsilon)}

Bodovani:
- Za drobnou chybu (napr. zapomenuty 1 prechod automatu) byly strzeny 2 body
- Pokud automat nepouzil pri prechodu do generovani nejake dalsi casti slova novy stav nebo novy symbol na zasobniku a tim padem umoznoval libolovne michani napr. pismene a a pismene b, bylo za to celkem 20 bodu, pokud byl automat jinak v poradku.
- Za pokus o vyreseni problemu konecnym automatem nebyl zadny bod

Podzim 2008 , 4.2.2009, Termín 04
Příklad 02

Vasim ukolem bylo dokazat pomoci Myhill-Nerodovy vety, ze jazyk

$L = \{w \in \{a,b\}^* \mid \#a(w) <= \#b(w) \}$

neni regularni.

Mohli jste ziskat az 35 bodu.

Spravne reseni mohlo vypadat nasledovne.
Predpokladejme ze jazyk L je regularni. Pak z M.N. vety plyne, ze relace $\sim_L$ ma konecny index. Tudiz existuje $n \in N$ takove, ze index $\sim_L$ je roven n (dale v textu a v pisemkach pouzivam zkratku $|\sim_L|$ pro index relace $\sim_L$, ackoliv to je spatny a nestandartni zapis a panu prednasejicimu se vubec nelibi :-)).
Uvazme n+1 slov:
$a,a^2,a^3,...,a^{n+1}$
Z predpokladu, ze $|\sim_L| = n$ dostavame, ze aspon dve slova z dane posloupnosti jsou v relaci. Nyni ukazeme, ze takova dve slova neexistuji (musime tedy uvazovat vsechny mozne dvojice slov).
Necht $1<=i,j<=n+1$ a $i!=j$. Bez ujmy na obecnosti predpokladejme, ze $i<j$.
Dale predpokladejme, ze $a^i \sim_L a^j$ (slova a^i a a^j jsou v relaci).
Pak dle definice prefixove ekvivalence ($\sim_L$) musi platit, ze:
$a^ib^i \in L <=> a^jb^i \in L$.
Ale a^ib^i patri do L a a^jb^i nepatri do L (jelikoz $i<j$) a tudiz nepalti, ze $a^i \sim_L a^j$.
Ukazali jsme tedy, ze v dane posloupnosti n+1 slov nejsou zadne dve slova v relaci a tudiz dostavame spor s predpokladme, ze $|\sim_L| = n$. Index $\sim_L$ je tedy nekonecno a z M.N. plyne, ze jazyk L neni regularni.

Hodnoceni:

Vase hodnoceni ve vetsine pripadu zalezelo na tom, jak spravne jste dokazali ze index relace $\sim_L$ je nekonecno.
Pokud jste uvedli ze index $\sim_L$ je nekonecno bez dukazu nebo s zcela spatny dukazem ziskali jste 5 bodu.
Pokud jste ve vasem dukazu pouze ukazali ze nejake dvojice slov nejsou v relaci, mohli jste ziskat 8 az 18 bodu.
Dalsi chyby byly mensi ci vetsi nepresnosti v dukazech ze index $\sim_L$ je nekonecno, ktere byly penalizovany 2 az 10 body.
Pokud jste se snazili zapsat tridy rozkladu podle $\sim_L$, ale zapomneli jste na slova obsahujici znaky 'b' dostali jste maximalne 18 bodu.
Pokud jste se jen snazili formulovat M.N. vetu, mohli jste ziskat maximalne 5 bodu (za nepresnosti pri formulaci M.N vety Vam bylo z techto 5 bodu jeste neco strzeno).
Castou chybou byla rovnez zamena prefixove ekvivalence ($\sim_L$) a libovolne prave kongruence ($\sim$).
Nektera Vase reseni byla hodne specificka a jejich hodnoceni jsem se snazil komentovat ve Vasich pisemkach.

Podzim 2008 , 4.2.2009, Termín 04
Příklad 03

V tomto příkladě jste měli pomocí DETERMINISTICKÉHO algoritmu pro syntaktickou analýzu určit, zda gramatika generuje dané slovo.

Ti, kdo si správně přečetli zadání a pracovali s deterministickým, tedy C-Y-K algoritmem, měli z větší části vyhráno. Tabulku sem vypisovat nebudu, mělo to vyjít, že gramatika slovo generuje. Při použití tohoto přístupu jste dostali 35 - 2*(počet spatně vyplněných políček v tabulce) bodů, tedy za každou chybu -2 body. Jako chybu jsem počítala jen políčka, která byla špatně vyplněna na základě údajů v nižších řádcích (ať už správně spočítaných nebo ne), takže chyby se nepropagovaly.

Ti, kdo si zadání přečetli špatně nebo nevěděli, jak vypadá deterministická analýza, a zkoušeli nedeterministickou analýzu, mohli dostat nejvýše 10 bodů.

Ti, kdo napsali jen přímé odvození slova, mohli dostat nejvýše 3 body.

Podzim 2008 , 4.2.2009, Termín 04
Příklad 04

jak jste si již jistě všimli, je od včerejška naskenován i čtvrtý list z posledního termínu. Pro úplnost přidávám komentář k bodování.

Vaším úkolem bylo zapsat **algoritmus, který k dané bezkontextové gramatice sestrojí ekvivalentní gramatiku bez epsilon-kroků**. V drtivé většině případů jste se snažili zformulovat algoritmus z přednášky (byť ne nutně přesně ve formě pseudokódu uvedeného na slidech/ve skriptech -- což pochopitelně nebylo třeba, důléžité bylo, aby byl zvolený popis dostatečně jednoznačný). Protože chyby, které se ve Vašich řešeních objevovaly byly dost různorodé, berte následující bodování jako orientační:

přesný popis výstupní gramatiky - 5 bodů

popsání množiny N_epsilon - 15 bodů (Zde malý komkentář - původně jsem Vám strhával sedm bodů, pokud jste explicitně nepopsali konstrukci této množiny, což by mělo být součástí korektního řešení. Protože však tato konstrukce není napsána ve slidech, po poradě s panem přednášejícím Strejčkem jsem písemky přebodoval. Pokud však někdo správně napsal, jaké neterminály mají být v této množině obsaženy a posléze naspal algoritmus, který napočítal něco jiného, body mu strženy byly.)

samotné "tělo" algorimu - 15 bodů
Zde lze těžko popsat nějakou "osnovu bodování", hodně záleželo na tom, zda byla vaše formulace algoritmu dostatečně jednoznačná. Obecně však bylo hodnocení mírné :).

Posledních 5 bodů bylo za správné vyřešení situace, kdy se kořen vstupní gramatiky objevil v množině N_epsilon. Pokud jste nenapsali, že neterminál S', který se v takovém případě stává kořenem výstupní gramatiky, je nově přidaný, strhával jsem 2 body.

Bodové srážky se většinou týkaly nejednoznačné, či přímo zmatené formulace algoritmu (typicky stylem "vynechám všechny možné kombinace neterminálů z N_epsilon"). Opravdu nestačí napsat příklad, na kterém postup ilustrujete. Jednotky bodů jsem rovněž strhával za formální nepřesnosti. Pokud však bylo ze zápisu patrné, že dobře chápete myšlenku algoritmu, ale pokulhává jeho formální zápis, alespoň polovinu bodů byste získat měli.

Podzim 2008 , 4.2.2009, Termín 04
Příklad 05

v tomto příkladu jste měli za úkol rozhodnout, zda existují zadané gramatiky, a toto rozhodnutí doložit buď napsáním oné gramatiky, nebo důkazem neexistence. Každá ze dvou podotázek byla hodnocena maximálně 15 body, z toho 5 bylo za správné určení odpovědi ANO/NE, zbytek za zdůvodnění. V případě špatné odpovědi jste mohli získat několik málo bodů za správné definice vlastní gramatiky a Chomského/Greibachové normální formy (max. 2 body za definici).

Správné odpovědi a časté chyby:
* CFG generující konečný jazyk, který není regulární, neexistuje. Každý konečný jazyk je regulární. Nejčastější chyba - napsali jste gramatiku, která byla bezkontextová a nebyla regulární, jazyk ale regulární byl.
* CFG generující kontextový jazyk existuje, je to libovolná CFG, protože každá CFG generuje bezkontextový jazyk a každý bezkontextový je i kontextový.
* CFG v Greibachové nebo Chomského normální formě, která není vlastní, existuje. Tady jste nejčastěji chybovali, tvrdili jste, že gramatika v GNF/CNF musí být vlastní, protože jste si pamatovali, že vstupem algoritmu pro převod do normálních forem je vlastní gramatika. V definici forem ale tato vlastnost požadována není, gramatika proto může obsahovat nepoužitelné symboly. Že to vyžaduje algoritmus je věc jiná, tam je to proto, že odstranění nepoužitelných symbolů je celkem snadné a může to gramatiku podstatně zjednodušit, což je pro převádění výhodné.

Podzim 2008 , 4.2.2009, Termín 04
Příklad 06

na tomto listu (ktery by mel byt naskenovan v pondeli) jste meli za ukol napsat nekolik definic a prokazat sve znalosti o uzaverovych vlastnostech bezkontextovych jazyku. Casti a) a b) byly stejne jako na predchozi pisemce a byly tedy i stejne bodovany (viz https://is.muni.cz/auth/diskuse/diskusni_forum_predmet.pl?guz=7262187;jn=v).

c) Definujte, kdy je neterminal v CFG levorekursivni (celkem 10b)

Neterminal A v CFG G = (N,\Sigma,P,S) se nazyva levorekursivni, pokud v G existuje derivace A =>^+ A\beta (kde \beta \in (N U \Sigma)^*)

- pro zisk plneho poctu bodu nebylo treba uvadet jak vypada \beta (to je jasne z kontextu), pokud jste tak ucinili a meli jste zde chybu -1b
- naopak symbol "+" je zde velmi podstatny a nelze jej na rozdil od definice v a) nahradit symbolem "" (protoze pak by byl _kazdy_ neterminal levorekursivni a definice by nemela prilis smysl) -- za tuto chybu -4b*
- v pripade, ze jste nejakym silne nevhodnym zpusobem do definice primichali pravidla (typicky jste derivaci nazvali pravidlem) -2b az -4b
- za dalsi pripadne drobne chyby -1b az -2b dle charakteru
- pokud jste v definici uvazili pouze primou rekursi, mohli jste ziskat az 3b

d) Uvedte 5 operaci, na ktere je trida bezkontextovych jazyku uzavrena (celkem 10b)

- ze zakladnich operaci to jsou sjednoceni, iterace, pozitivni iterace, zretezeni a prunik s regularnim jazykem
- dale se objevilo: substituce, homomorfismus, zrcadlovy obraz (uznavana i reverse), mocnina
- CHYBNE ODPOVEDI: prunik, rozdil, doplnek, symetricky rozdil

- za kazdou spravnou odpoved +2b
- za kazdou spatnou odpoved -2b
- za nejasne/castecne spravne popisy operaci byly ignorovany nebo ohodnoceny 1b v zavislosti na konktetnim pripadu
- pokud jste meli alespon jednu spravnou odpoved, tak jste ziskali alespon 1b

Podzim 2009 , 7.1.2010, Termín 01

O každém z následujících jazyků rozhodněte, zda je bezkontextový. Svá tvrzení dokažte. *(Pro důkaz, že jazyk je bezkontextový, stačí napsat odpovídající gramatiku nebo automat.)*

Příklad 1
50 bodů

(a) $L_1 = \{w \in \{a,b,c\}^* \mid \#_a(w) + \#_b(w) \leq \#_c(w)\}$

(b) $L_2 = \{w \in \{a,b,c\}^* \mid \#_a(w) \leq \#_b(w) \leq \#_c(w)\}$

Jazyk {w ∈ {a,b,c}* | #a(w) + #b(w) ≤ #c(w)} bezkontextový byl. Dokázat jste to mohli sestrojením zásobníkového automatu nebo bezkontextové gramatiky. Za rozhodnutí, že se jedná o bezkontextový jazyk jste dostali 5 bodů, zbylých 20 bodů jste pak mohli dostat za samotnou konstrukci.

A = ({Q,Qacc},{a,b,c},{Z,A,C},δ,Q,Z,{Qacc})

kde:

δ(Q,a,Z)={(Q,AZ)}	δ(Q,c,A)={(Q,ε)}
δ(Q,b,Z)={(Q,AZ)}	δ(Q,c,C)={(Q,CC)}
δ(Q,c,Z)={(Q,CZ)}	δ(Q,a,C)={(Q,ε)}
δ(Q,a,A)={(Q,AA)}	δ(Q,b,C)={(Q,ε)}
δ(Q,b,A)={(Q,AA)}	

δ(Q,ε,Z)={(Qacc,ε)}
δ(Q,ε,C)={(Qacc,ε)}

Nejčastější chybou při konstrukci automatu bylo to, že automat počítal pouze o kolik více znaků a/b přečetl oproti znakům c, ovšem pokud přečetl více znaků c, již je nepočítal. To vedlo k tomu, že nebyla akceptována slova jako cccaaa, protože automat na céčka na začátku "zapoměl". Za takový automat jste mohli dostat nejvýše 8 bodů. Často se vyskytovaly také různé formální chyby jako chybějící/přebývající množinové závorky -- podle četnosti se za ně strhávaly 1-3 body.

Přestože dost z vás konstruovalo gramatiku, která měla generovat daný jazyk, správné řešení se objevilo jen jedno. Nejčastěji byla k vidění gramatika podobná následující:
({S,A,C},{a,b,c},P,S), P = {
S -> ε | ACS | CAS | ASC | CSA | SAC | SCA,
A -> ε | a | b,
C -> c }
Touto gramatikou nelze generovat slova, jako je například aaccccaa (proč?). Byla tedy hodnocena maximálně 12 body.

Druhý jazyk, {w ∈ {a,b,c}* | #a(w) ≤ #b(w) ≤ #c(w)}, bezkontextový nebyl. Dokázat jste to mohli pomocí lemmatu o vkládání pro bezkontextové jazyky. Za odpověď ano/ne 5 bodů, dalších až 20 pak za samotný důkaz.

slovo je a^n b^n c^n
vwx se skládá pouze z jednoho typu písmen, tj. 3 možnosti: a^i, b^i, c^i, kde 0<=i<=n

vwx se skládá z dvou typů písmen, tj. 2 možnosti: a^j.b^k, b^j.c^k, kde 0<j<n, 0<k<n

vwx se nemůže skládat z více typů písmen, protože jakákoli část slova z=a^n.b^n.c^n délky n obsahuje maximálně 2 typy písmen (jinými slovy, slovo z můžeme zapsat jako pqr, kde pokud je q délky nejvýše n, pak buď p nebo r je délky alespoň n, tedy alespoň jedna z těchto částí obsahuje alespoň n znaků - zleva (a^n) ... pokud je délka p alespoň n zprava (c^n) ... pokud je délka r alespoň n prostřední část tedy nemůže obsahovat všechny 3 typy znaků.

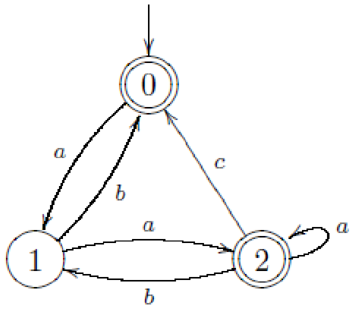
vwx nemůže být prázdné z definice PL
žádná jiná možnost není

Při důkazu obměnou PL byl nejčastěji problém s vynecháním některého možného rozdělení slova. Častou chybou byl také nesprávný zápis těchto rozdělení. Podle závažnosti jste za takový důkaz mohli dostat od 4 do 12 bodů.

Podzim 2009 , 7.1.2010, Termín 01

Je dán automat A:

Příklad 2
30 bodů



Napište regulární výraz E popisující jazyk $L(A)$.
(Rovnost $L(E) = L(A)$ nemusíte dokazovat, pokud použijete standardní algoritmus a zakreslíte všechny jeho mezivýsledky.)

1. Použití standardního algoritmu

Výsledný RE závisel na tom, v jakém pořadí jste odstraňovali původní stavy. Ideální bylo odstraňovat stavy v pořadí 1-2-0, v takovém případě totiž docházelo k nejmenšímu narůstání délky RE u jednotlivých hran, čímž se rovněž nižovala pravděpodobnost, že uděláte chybu.

Body jsem rozděloval následovně:
Správná inicializace algoritmu (přidání nového počátečního a koncového stavu a epsilon-přechodů): 3 body (ke správnému výsledku jste v jistém případě mohli dojít i bez přidání nového poč. stavu, v takovém případě jsem body nestrhával).

Za správné odstranění každého původního stavu: 8 bodů (celkem tedy 24 bodů). V případě, že jste při odstraňování stavu zapomněli na nějakou hranu, strhával jsem v příslušném kroku 2 až 8 bodů (podle počtu těchto chyb).

Za zapsání výsledku: 3 body. Tohle byl spíš takový bonus, implicitně jsem bral jako výsledný ten RE, který byl návěštím poslední hrany reg. přech. grafu po ukončení algoritmu.

Někdy Váš postup připomínal standardní algoritmus, ovšem v některých zásadních aspektech se lišil, často jste používali zcela nekorektní kroky. V takovém případě bodování probíhalo podle následujícího odstavce.

2. Použití jiného postupu
Zde bylo 20 bodů za RE a 10 bodů za důkaz/zdůvodnění. Vhodné zdůvodnění nenapsal nikdo, většinou jste se do této části dostali tak, že jste použili nějakou zmatenou (a nekorektní) alternativu standardního algoritmu. Pár pokusů o zdůvodnění jsem našel, ty však byly zcela mimo. Správný regulární výraz nenapsal (a tedy plných 20 bodů nezískal) bohužel také nikdo. Pokud jste si však dali pozor na to, aby Vámi zadaný výraz popisoval nějakou (nekonečnou) podmnožinu jazyka L(A), dostali jste 12 bodů.

Byl jsem připraven dát 16 bodů tomu, pro jehož RE by platilo L(A)⊆L(E), byl-li by (fuj) jazyk L(E) dostatečně netriviální. (Mám za to, že postihnout daným výrazem všechna slova jazyka L(A) je obtížnější, než postupně přidávat slova, která automat převedou do konc. stavu.) Nikdo takový se však nenašel.

Pokud v jazyce popsaným Vaším RE chybělo nekonečně mnoho slov z jazyka L(A) a zároveň v něm přebývalo nekonečně mnoho slov z co-L(A), žádný bod jste bohužel nezískali.

Globální hodnocení:
Nejvýše dva body jsem strhával, pokud jste ve výsledku neuvedli RE (sjednotítka namísto +, použití pozitivní iterace apod.), dále jsem odebíral až čtyři body za zmatené uzávorkování výsledku (pokud byl předchozí výpočet víceméně správný). V několika extrémních (a nula body ohodnocených) případech jsem kvůli zmatenému uzávorkování a chybějícímu zdůvodnění korektnosti nebyl vůbec schopen rozeznat, co má být vlastně výsledek, tak si na to prosím příště dávejte pozor. Výsledný počet bodů pochopitelně závisí na kombinaci nalezených chyb.

Drtivá většina z Vás získala alespoň 20 bodů, většina chyb byly chyby z nepozornosti. Někteří nepostupovali striktně podle algoritmu, ale body jsem jim nestrhával, neboť jejich modifikace neovlivnila korektnost výsledku. Například jste při odstraňování stavu se smyčkou nejprve k RE zapsanému u této smyčky "přičetli" prázdné slovo a až následně jste tento výraz iterovali. To sice není chybné, ale je to zcela zbytečné a zbytečně se tak komplikuje výsledný RE. Rovněž mějte na paměti, že nové hrany přidáváme výhradně tehdy, pokud odebíráme nějaký stav, jiný postup (hlavně pak často viděná kombinace odebírání některých hran a přidávání jiných nových hran, aniž by se zároveň odstranil nějaký stav) může vést k nekorektnímu výsledku a k nula bodům z příkladu. :(

Podzim 2009 , 7.1.2010, Termín 01

Je dána gramatika $\mathcal{G} = (\{S, A, B\}, \{a, b, c\}, P, S)$, kde

Příklad 3
35 bodů

$$
\begin{aligned}
P = \{\ S &\rightarrow ab \mid Ab \\
A &\rightarrow BaA \mid Sc \mid Ac \\
B &\rightarrow AA \mid bB\ \}.
\end{aligned}
$$

Převeďte gramatiku $\mathcal{G}$ na ekvivalentní nelevorekursivní bezkontextovou gramatiku. Pokud nepoužijete algoritmus z přednášky, dokažte ekvivalenci výsledné gramatiky s gramatikou ze zadání.

Ospravedlňujem sa, ale pre vlastnú obrovskú hlúposť som obsah tohoto príspevku nechtiac premazal a nemám zálohu. Nebudem už rozbor písať znovu, ale skúsim ho získať od IStechnikov.

Podzim 2009 , 7.1.2010, Termín 01

Napište algoritmus, který pro zadanou redukovanou bezkontextovou gramatiku $\mathcal{G} = (N, \Sigma, P, S)$ spočítá množinu M všech neterminálů, z kterých lze odvodit neprázdný řetězec, tj. $M = \{A \in N \mid A \Rightarrow^* w \text{ pro nějaké } w \in \Sigma^+\}$.

Příklad 4
40 bodů

Řešení:

```
i := 0
N_0 := \emptyset
repeat
  i := i + 1
  N_i := N_{i - 1} U
              {A | \exists A -> \aplha :
                  \alpha = (N U \Sigma)*.
                               (N_{i - 1} U \Sigma).
                               (N U \Sigma)* }
until N_i = N_{i - 1}
M := N_i
```

Gramatika byla redukovaná, takže do množiny M určite patří všechny neterminály A takové, že existuje pravidlo A -> \aplha, kde \alpha obsahuje terminál. Všechny takové neterminály bude obsahovat N_1.

Do N_i se potom přidají neterminály A takové, že existuje pravidlo A -> \aplha, kde \alpha obsahuje nějaký neterminál z N_{i - 1}. Opět se zde využívá toho, že gramatika je redukovaná.

Hodnocení:

Drtivá vetšina z vás vzala algoritmus na počítání množiny neterminálů z nichž lze vygenerovat terminální řetěz a nahradila v něm iteraci za pozitivní iteraci, což ovšem není dobře, protože tento postup např. pro gramatiku:

S -> AB
A -> \epsilon
B -> b

nepřidá do množiny M neterminál S, ze kterého ale zjevně neprázdné slovo vygenerovat lze, totiž "b". Nekteří z vás ten algoritmus ani neupravovali a nechali tam tu iteraci. Tak jako tak, kdo prokázal znalost tohoto algoritmu, dostal 20 bodů. Kdo v něm udělal nějaké drobnější chyby, dostal 15 bodů, a kdo v něm udělal závažnější chyby, např. pro druhou a další iteraci uvažoval pouze pravidla, která mají na pravé straně jenom neterminály, dostal 10 bodů.

Kdo vypočítal množinu M jako N \setminus N_\epsilon, kde N_\epsilon jsou neterminály, ze kterých lze vygenerovat \epsilon, dostal 5 bodů.

5 bodů dostal také ten, kdo se pokusil o algoritmus, který sice vůbec nefungoval, ale byla v něm alespoň nějaká myšlenka.

Podzim 2009 , 7.1.2010, Termín 01

Rozhodněte, zda existují následující gramatiky. V kladném případě uveďte příklad takové gramatiky, v záporném důkaz její neexistence.

Příklad 5
15+15 bodů

(a) Regulární gramatika, která je zároveň bezkontextovou gramatikou v CNF.

(b) Bezkontextová gramatika v GNF, která je cyklická.

Řešení:
(a) G = ({S},{a},{S -> a}, S)
(b) Taková gramatika neexistuje. Gramatika v GNF má každé pravidlo začínající terminálem, proto každá větná forma, která vznikne z jednoho neterminálu v alespoň jednom kroku, musí začínat terminálem. Žádná gramatika v GNF tedy nemůže povolovat odvození tvaru A =>+ A.

Hodnocení:
5 bodů za odpověď ano/ne, 10 za gramatiku (a) / důkaz (b)
(a) za gramatiku, která splňovala alespoň jednu z podmínek 3b.
(b) vaše (pokusy o) důkazy se daly zhruba rozdělit do následujících skupin:
Argument vstupem algoritmu - *vstup algoritmu pro převod do GNF požaduje acyklickou gramatiku, proto je každá gramatika v GNF acyklická. Hodnocení: 4b.*
Argument algoritmem - *vstup algoritmu požaduje acyklickou gramatiku a během algoritmu se acykličnost gramatiky nezmění. Hodnocení: 5b.*
Argument jednoduchými pravidly - *gramatika v GNF nemá jednoduchá pravidla, proto není cyklická. Hodnocení: 6b.*
Argument GNF je vlastní. - *gramatika v GNF je vždycky vlastní, proto je acyklická. Hodnocení: 7b.*
Neúplný důkaz. - *gramatika v GNF má pravidla tvaru ..., a proto je acyklická. Hodnocení: 8b.*
Správný důkaz, varianta I. *Viz výše. 10b.*
Správný důkaz, varianta II. *Gramatika v GNF nemá jednoduchá a epsilon pravidla, což jsou jediné dva způsoby jak dostat odvození A =>+ A. Hodnocení: 10b.*
Za různé formální chyby a nepravdivá tvrzení: -1b za kus.
Několik lidí napsalo, že kontextová gramatika je regulární. To jste jako v učení se nedošli za prvních pár stránek skript?

Podzim 2009 , 7.1.2010, Termín 01

Příklad 6
30+10 bodů

(a) Definujte pojmy *gramatika* a *kontextová gramatika.*

(b) Definujte, kdy má bezkontextová gramatika *vlastnost sebevložení.*

Řešení: **(a)** Gramatika je čtveřice (N, Σ, P, S) [2b, 2b za správný formální zápis, tj. závorky] taková, že:

- N je konečná (a neprázdná) [2b] množina neterminálů [2b],
- Σje konečná [2b] množina terminálů [2b] taková, že $N \cap \Sigma = \emptyset$[2b]
- $P \subseteq (N \cup \Sigma)^* N (N \cup \Sigma)^* \times (N \cup \Sigma)^*$[2b] je množina pravidel [2b] a
- S je počáteční neterminál [2b]

Gramatika je kontextová, pokud pro všechna její pravidla platí, že levá strana je nejvýše tak dlouhá jako pravá strana [6b] s případnou výjimkou pravidla S -> epsilon [2b], pokud se S nevyskytuje na žádné pravé straně pravidel [2b].

(b) CFG má vlastnost sebevložení, pokud existuje neterminál A takový, že existuje odvození A =>* uAv, kde u, v $\in \Sigma^+$.

Hodnocení: U (a) zřejmé s předchozího, -1b za nepravidvá tvrzení a formální chyby (jiné než závorky u čtveřice gramatiky). Za definici "kontextová gramatika je ta, která generuje kontextový jazyk" byly max 3 body z deseti.
U (b) byly typické chyby: špatný typ u, v (nesmí být prázdné, nesmí obsahovat neterminály) -2b, počáteční neterminál místo libovolný neterminál -2b, pokud někdo napsal, že gramatika musí obsahovat pravidlo A -> uAv, mohl dostat max 4 body, různá nepravidvá tvrzení a formální chyby po -1 bodu.
Mnozí si pořád pletou pravidla a odvození, -> a =>. Za to byly taky -2b.

Podzim 2009, 13.1.2010, Termín 02

Navrhněte bezkontextovou gramatiku $\mathcal{G}$ v Greibachové normální formě generující jazyk

Příklad 1
40 bodů

$$L(\mathcal{G}) = \{a^i b^j c^k d^{2j} \mid i, j, k > 0\}.$$

(Rovnost $L = L(\mathcal{G})$ není třeba dokazovat.)

3b

Správnym riešením je napríklad nasledujúca gramatika:
G=({S,A,B,C,D},{a,b,c,d},P,S), kde
P={ S -> aAB | aB,
A -> aA | a,
B -> bBDD | bCDD,
C -> cC | c,
D -> d }

Za správne riešenie bolo možné získať až 40 bodov. Je to pomerne veľa za taký krátky príklad a preto sa aj bodové zrážky zdajú na prvý pohľad trochu vysoké.
Za chýbajúcu "hlavičku" som strhával 10b, za neoznačenie množiny pravidiel 2b, rôzne formálne nezrovnalosti v zápise 1-2b.
2-5b som strhával za rôzne chýbajúce pravidlá. Napr. 5b šlo dole, ak ste si nedali pozor a vaša gramatika generovala aj slová bez b-čiek a d-čiek, napríklad slovo ac.
Ak ste uviedli gramatiku, ktorá nebola v GNF, tak sa v podstate vaše body delili dvoma. Teda ste mohli maximálne získať 20b. Neplatilo to v prípade, ak bola uvedená gramatika v podstate dobre až na jedno-dve pravidlá. (Napríklad, ak by gramatika hore mala pravidlo S -> AB, alebo ak by nemala neterminál D.) Vtedy som za takúto chybu strhol 10b.
Za gramatiku generujúcu jazyk $\{a^i b^{2j} c^k d^j \mid i,j,k > 0\}$ *ste prišli o 8 bodov (o 10, ak ste "zviazali" napr. b-čka a c-čka). Pri iných jazykoch som sa snažil zistiť, aký jazyk vaša gramatika vlastne generuje a podľa toho nejak adekvátne prideliť body. Nie vždy to bolo úplne jednoduché. Častými generovanými jazykmi boli regulárne jazyky* $\{a^i b^j c^k d^l \mid i,j,k,l > 0\}$ *a* $\{a^i b^j c^k d^{2l} \mid i,j,k,l > 0\}$*, prípadne dokonca* $\{a,b,c,d\}^*$.
Mnohí z vás sa riadili zásadou: "Načo to robiť jednoducho, keď sa to dá zložito". A potom navrhovali zbytočne zložité gramatiky, prípadne gramatiky, ktoré neboli v GNF a potom ich pracne prevádzali do GNF (taký postup je legitímny, bohužiaľ mnohí sa v tom procese zmýlili), odstraňovali ľavú rekurziu, tam kde nebola a tak podobne.

Podzim 2009, 13.1.2010, Termín 02

Zkonstruujte regulární výrazy popisující následující jazyky:

Příklad 2
36 bodů

(a) $L_1 = \{w \in \{0,1\}^* \mid w \text{ neobsahuje podslovo } 11\}$

(b) $L_2 = \{w \in \{a,b,c\}^* \mid w \text{ obsahuje podslova } aaa \text{ a } abc\}$

Riesenie:

a) Ulohu bolo idealne riesit pomocou konecneho automatu, minimalny mal v tomto pripade 3 stavy, z nich dva koncove a jeden, ktory neviedol do ziadneho koncoveho stavu (peklo, pre zuplneniu prechodovej funkcie). Upravou na regularny vyraz, podla algoritmu z prednasky, dostavame vyraz
(0+1.0)^*.(1+epsilon)

Problem sa dal samozrejme riesit aj bez pomoci minimalneho automatu, no riesenie nebolo tak pekne viditelne, co viedlo k viacerym chybam.

Hodnotenie:
- pri rieseni automatom:
- za ciastocne spravny automat do 8 bodov
- 9 bodov za spravny automat
- odvodzovanie regularneho vyrazu bolo tak na 3 kroky, teda za kazdy zly -3 body, alebo podla zavaznosti chyby
- pri rieseni bez automatu:
- ak popisany jazyk nebol ani podmnozinou L_1 0 bodov
- ak bol podmnozinou, zhruba 9 bodov, na zaklade toho aku velku cast jazyka ste popisali, napr ak chybalo iba 0^* 15-16 bodov, ak chybali aj slova konciace alebo zacinajuce 1 cca 12 bodov, a pod.
- ak vyraz nebol regularny (co sa stavalo relativne casto) -2 az -4 body

b) Uloha sa dala lahko intuitivne riesit, najkrajsi zapis je
(a+b+c)^*.aaa.(a+b+c)^*.abc.(a+b+c)^* + (a+b+c)^*.abc.(a+b+c)^*.aaa.(a+b+c)^*
+ (a+b+c)^*.aaabc.(a+b+c)^*
Mnohi to opat riesili automatom, a to tak, ze minimalny automat pre tento jazyk vytvorili ako sucin automatov pre jazyky slov obsahujucich jednotlive podslova. To je samozrejme tiez spravny postup, no viedol k siahlodlhym upravam a vysledkom, v podstate necitatelnym a skoro vzdy nespravnym...

Hodnotenie:
- v pripade pokusu o priamy zapis:
- kazda cast vyssie uvedeneho suctu bola hodnotena 6 bodmi
- v pripade zleho zapisu napr. (abc)^* miesto (a+b+c)^* cca -2 body, resp. podla zavaznosti chyby
- vo viacerych pripadoch sa stalo, ze ste uvazovali slova, ktore obsahuju podslovo aaa ALEBO podslovo abc, co samozrejme nie je L_2, 0 bodov
- pri rieseni automatom:
- ak spravny automat, ale bez pokusov o upravu na regularny vyraz 6 bodov
- spravny automat a par krokov 9 bodov
- dalej sa vacsinou nikto nedostal, ak ano body podla spravnosti...
- mnohi opat nevedeli co vlastne regularny vyraz je -2 az -4 bodov podla zavaznosti

Podzim 2009, 13.1.2010, Termín 02

Je dána gramatika $\mathcal{G} = (\{S, A, B\}, \{a, b\}, P, S)$, kde

Příklad 3
20+15 bodů

$$
\begin{aligned}
P = \{\ S &\rightarrow aAA \mid Bb \\
A &\rightarrow aB \mid bb \\
B &\rightarrow Bb \mid bS \mid \varepsilon\ \}.
\end{aligned}
$$

(a) Zkonstruujte rozšířený PDA $\mathcal{A}$ pro nedeterministickou syntaktickou analýzu zdola nahoru. Uveďte způsob akceptování.

(b) Zapište akceptující výpočet automatu $\mathcal{A}$ nad slovem $aabbb$.

prázdným zásobníkem

ad a)
Za spravne reseni jste mohli ziskat 20 bodu:
- za spravna redukcni pravidla - 5 bodu
- za spravna cteci pravidla - 5 bodu
- za spravne akceptovani 5 bodu

Poslednich 5 bodu bylo za hlavicku PDA a za formalni korektnost.

Zde jsem typicky strhaval:
- za formalni chyby (spatna hlavicka PDA, chybejici zavorky a pod) - srazka 1 az 3 body
- za dvoji definici pravidla delta(q,epsilon,Bb) - srazka 2 body

ad b)
Za spravne reseni jste mohli ziskat 15 bodu.

Za drobny nedostatek, ktery vsak nevedl ke spatnemu vypoctu 10 az 13 bodu.
Za spatny vypocet obsahujici zavaznou chybu 5 bodu.
Za spatny vypocet obsahujici vicero zavaznych chyb 3 body.
Za uvedeni aspon jednoho spravneho kroku vypoctu 2 body.

Za absolutne spravny PDA pro syntaktickou analyzu shora dolu jste mohli ziskat maximalne 17.

Podzim 2009, 13.1.2010, Termín 02

Napište algoritmus, který pro zadanou redukovanou bezkontextovou gramatiku $\mathcal{G} = (N, \Sigma, P, S)$ a zadaný symbol $a \in \Sigma$ spočítá množinu M všech neterminálů, z kterých lze odvodit řetězec obsahující a, tj.

Příklad 4
40 bodů

$$M = \{A \in N \mid A \Rightarrow^* uav \text{ pro nějaké } u, v \in \Sigma^*\}.$$

Napíšte algoritmus, ktorý pre redukovanú bezkontextovú gramatiku $G = (N, \Sigma, P, S)$spočíta množinu M tých neterminálov, z ktorých sa dokáže odvodiť slovo terminálov obsahujúce $a \in \Sigma$. Formálne:

$$M = \{A \in N | A \Rightarrow^* \alpha a \beta; \alpha, \beta \in \Sigma^*\}$$

Ukážka riešenia:

  A_0 <- ∅
  i <- 0
  repeat
      i <- i + 1
      A_i <- A_{i-1} ∪ {B ∈ N | B -> αXβ; α,β ∈V^*, X∈({a}∪A_{i-1})}
  until A_i = A_{i-1}
  M <- A_i

Algoritmus funguje preto, lebo vstupná gramatika je redukovaná. Preto každé pravidlo je použiteľné a z každého konečného slova nad Vsa v konečnom čase dá odvodiť konečné slovo nad Σ.

Hodnotenie a časté chyby:
Najviac ste za štvrtý list mohli dostať 40 bodov. Príklad sa dosť podobal tomu z predchádzajúceho termínu, preto aj dopadol o poznanie lepšie. Veľká časť z vás dokázala prísť s fixpoint algoritmom, aj keď už nebol tak často dotiahnutý do konca. Bodovanie vzdialene pripomína to minulé a tentokrát prebiehalo tak, že riešenie dostalo základný počet bodov podľa toho, do ktorej z nasledovných kategórií sa najviac hodilo. Od toho sa odpočítali menšie zrážky (zriedkavejšie pripočítali bonusy) rádovo v jednotkách bodov podľa malých chýb či rôznych detailov.

  40 bodov: správne riešenie
  30 bodov: správne naznačené riešenie problému a myšlienka, neúplná funkčnosť na všetkých vstupoch
  25 bodov: správne naznačené riešenie, zlá funkčnosť pre väčšinu vstupov
  20 bodov: použitý fixpoint algoritmus na spočítanie neterminálov
  15 bodov: správne napočítaná množina zodpovedajúca 1. iterácii vzorového algoritmu
  5 bodov: ľubovoľný algorimus, ktorý má aspoň nejakú myšlienku

Najčastejšie chyby sú zhrnuté v tomto algoritme:

  A_0 <- {B ∈ N | B -> αaβ; α,β ∈ Σ^*}
  i <- 0
  repeat
      i <- i + 1
      A_i <- A_{i-1} ∪ {B ∈ N | B -> αXβ; α,β ∈Σ^*, X ∈A_{i-1}}
      / alebo v nie oveľa lepších prípadoch: */*
      A_i <- A_{i-1} ∪ {B ∈ N | B -> αXβ; α,β ∈(Σ∪A_{i-1})*, X ∈A_{i-1}}
      A_i <- A_{i-1} ∪ {B ∈ N | B -> X; X ∈(Σ∪A_{i-1})*}
  until A_i = A_{i-1}
  M <- A_i

Nájdenie zaujímavých gramatík, pre ktoré tieto postupy vrátia a nevrátia správny výsledok, vám nechávam ako cvičenie (Ak si chcete overiť váš tip, píšte ho do odpovedí bielou farbou).
Pár všeobecných rád na dobrú noc:
Čo sa týka zápisu algoritmu, neodpustím si kritiku niektorých riešení, konkrétne tých, ktoré sa pokúšali o implementáciu v céčku. I napriek tomu, že som príslušný kurz neabsolvoval, som zápis pochopil - v tom problém nebol. Ale iterovanie nad množinou neterminálov pomocou for (i = 0; i++; i < |N|) { N[i] = ... } mi príde ako dosť nepraktické. Nebojte sa používať pseudokód, ušetríte tak dosť času a potenciálnych chybičiek, ktoré sa takto dajú spraviť oveľa jednoduchšie (inak sa ale za to body pochopiteľne nestrhávali). K zápisu algoritmu odporúčam uviesť slovný popis, môže zachrániť pár bodíkov pri chybách vo formálnom zápise, ktoré budú hodnotené ako preklep a nie ako vecná chyba. Naopak, slovný popis bez algoritmu je síce akceptovateľný, ale u mnohých bol až príliš viacznačný na to, aby bol zaradený do najvyšších kategórií.
Na záver upozorňujem tiež, že niektorí si zobrali moje napomenutie k stručnosti odminula možno až príliš k srdcu. Rozhodne sa neoplatí odovzdávať prázdny papier! Aj myšlienky idúce správnym smerom vám získajú pár bodov, pokiaľ ich zvládnete na papier nejak preniesť. Ďakujem aspoň tým, ktorí sa rozhodli pri nedostatku relevantných myšlienok nakresliť aspoň pekný obrázok, prispeli k dobrej nálade opravujúceho a teda následne aj k zvýšeniu priemerného počtu bodov príkladu.

Podzim 2009, 13.1.2010, Termín 02

Rozhodněte, zda existují následující gramatiky. V kladném případě uveďte příklad takové gramatiky, v záporném důkaz její neexistence.

Příklad 5
15+15 bodů

2 (a) Bezkontextová vlastní gramatika s jediným neterminálem, která generuje nekonečný jazyk obsahující i slovo ε.

2 (b) Kontextová gramatika, která není bezkontextová, ale generuje regulární jazyk.

Řešení:
(a) NE, neexistuje. Důkaz: Nechť G = ({S},E,P,S) je bezkontextová vlastní gramatika s jedinným neterminálem. Pokud má obsahovat slovo \eps, pak musí nutně obsahovat pravidlo S -> \eps a S neobsahovat na pravé straně žádného pravidla. Protože je ale S jedinný neterminál, budou na pravých stranách pravidel pouze řetězce terminálů a jazyk tudíž konečný (P je vždy konečná), což je ve sporu s požadovanou nekonečností jazyka.
(b) ANO, existuje. Např. G = ({S,A},{a},P,S), kde P = {S -> AA, AA -> aa}.

Hodnocení:
** 5 bodů za správnou odpověď ano/ne, 10 za důkaz (a) / gramatiku (b)*
** 1-3 body při chybné odpovědi ano/ne, ale prokázání relevantní znalosti*

(a) Přiměřený počet bodů podle úrovně důkazu. Výrazné srážky za argumentování typu: "Gramatika je necyklycká, proto nemůže generovat nekonečný jazyk", "Gramatika neobsahuje epsilon pravidla, proto nemůže generovat slovo \eps."

(b) Z 10 bodů za gramatiku bylo 7 bodů za pravidla, 3 body za popis G = (N,E,P,S). Stržení 1-3 bodů za drobné porušení podmínek kontextovosti.

Rozumím, že se může zdát stržení 3 bodů při neuvedení N,E,P,S přísné. Byl k tomu ale pádný důvod. Gramatika totiž byla kontextová a velké množství těch, kteří část G = (N,E,P,S) uvedli, to nezvládlo bez chyb. Tou nejčastější bylo uvedení G = ({S,AA},{a},P,S) namísto správného v (b) výše. Upozorňuji, že tím z gramatiky dokonce uděláte bezkontextovou, protože interpretujete AA jako jeden neterminál.

Podzim 2009, 13.1.2010, Termín 02

Příklad 6
10+10+10+14 bodů

(a) Definujte, kdy má bezkontextová gramatika *vlastnost sebevložení.*

(b) Nechť L je jazyk nad abecedou Σ. Definujte relaci $\sim_L$ zvanou *prefixová ekvivalence pro L.*

(c) Definujte, kdy je neterminál bezkontextové gramatiky *levorekursivní.*

16

(d) Napište 5 operací nad jazyky, na které je třída bezkontextových jazyků uzavřená. Dále napište 2 operace nad jazyky, na které třída bezkontextových jazyků není uzavřená.

a) *Řešení:* CFG má vlastnost sebevložení, pokud existuje neterminál A, takový, že existuje odvození A =>* uAv, kde u, v jsou neprázdné řetězce terminálů.

Hodnocení: *celkem 10 bodů, -2 za nesprávný typ u, v, -1/-2 za různé formální chyby (opět připomínám - pravidlo není odvození, -> není =>). Když někdo napsal, že gramatika musí obsahovat pravidlo A -> uAv, dostal max 4 body.*

(b) Relace $\sim_L$ je definována takto: $u \sim_L v \overset{def}{\Longleftrightarrow} (\forall w \in \Sigma^* : uw \in L \iff vw \in L)$.

Hodnocení: *celkem 10 bodů, typické chyby byly: špatný kvantifikátor u w (-3b), špatná logická spojka v závorce (místo ekvivalence konjunkce apod., -2b), kvantifikace přes w na začátku definice (-2 body), špatný typ w (např. $w \in \Sigma^+$, -2 body), apod. (Některé chyby dohromady byly penalizovány míň, než součet jejich penazlizací.) Pokud místo definice prefixové ekvivalence napsal někdo definici vlastnosti pravé kongruence, dostal 2 body za snahu, ale jen když byla definice úplně dobře.*

(c) *Řešení:* Neterminál A bezkontextové gramatiky je levorekursivní, pokud existuje odvození $A \Rightarrow^+ A\alpha$, kde $\alpha \in (\Sigma \cup N)^*$.

Hodnocení: *celkem 10 bodů, -3 body za odvození v libovolném počtu kroků (musí být alespoň v jednom kroku), -2 za špatný typ α, pokud místo levé rekurse někdo definoval jenom přímou, dostal max. 3b.*

(d) *Řešení:* Třída CFG je uzavřená na: sjednocení, zřetězení, iteraci, pozitivní iteraci, průnik s regulárním jazykem, (reverzní obraz, mocninu, substituci, homomorfismus, inverzní homomorfismus, ...)
Třída CFG není uzavřená na: průnik, doplněk, (rozdíl, ...)

Hodnocení: *2 body za správnou operaci, -1 za špatnou, 0 za žádnou nebo nesmyslnou (viz níže). Pokud u některé části bylo napsáno víc operací, než bylo třeba, a některé z nich byly špatné, počítal jsem -2 za každou špatnou nad potřebný počet. (Př.: Místo 5 operací někdo napsal 7, z nichž byly tři špatné, hodnocení 4*2 - 1 - 2 - 2 = 3 body.)*

Co se týče nesmyslných operací, během opravování jsem se dozvěděl, že třída CFG je uzavřená na: součet (co to je?), kartézský součin (to není operace na jazycích), podíl (zase netuším, co to je), doplněk s regulárním jazykem (dtto), na pravou kongruenci a na prefixovou ekvivalenci (to jsou relace, ne operace). A když už jsem u toho, slovo "třída" je ženského rodu a předložka "na" ve spojení "uzavřená na" se pojí se čtvrtým pádem, takže: "Třída není uzavřená na doplněk." a nikoli: "Třída není uzavřený na doplňku."

Podzim 2009, 25.1.2010, Termín 03

Navrhněte zásobníkový automat $\mathcal{M}$ akceptující jazyk

Příklad 1
40 bodů

$$L = \{a^i b^{2j} c^k d^i \mid i, j, k > 0,\ j > k\}.$$

Uveďte, jakým způsobem navržený automat akceptuje.

Vaším úkolem bylo navrhnout ZA pro jazyk a^ib^{2j}c^kd^i, kde i,j,k > 0 a j>k.
Správný postup mohl vypadat zhruba takto:
* automat načítá symboly A, ukládá na zásobník
* pokud je na zásobníku alespoň jedno A a na vstupu b, přejde do dalšího stavu pro načítání b; pro b je vhodné mít dva stavy, abyste mohli sledovat liché a sudé symboly (ale není to nutné, lze to řešit i přes různé symboly na zásobníku); v našem postupu za každé druhé B přidáme na zásobník B
* pokud byl přečten sudý a nenulový počet b a na vstupu je c, přejde se do dalšího stavu a odebere se B ze zásobníku; v tomto stavu se dále za každé c odebere B
* další přechod přejde pod epsilon do nového stavu a smaže B ze zásobníku; v tomto stavu se dále může pod epsilon odebírat B
* je-li na zásobníku A a na vstupu d, přejde se do posledního stavu, kde se odebírají ze zásobníku A za každé d na vstupu
* akceptuje se prázdným zásobníkem

Klíčové vlastnosti slov a odpovídající hodnocení schopnosti Vašeho ZA tyto vlastnosti "kontrolovat":
** správné pořadí znaků: 5b*
** i,j,k > 0: 7b*
** j > k: 8b*
** počet a = počet d: 5b*
** sudý počet b ve slově: 10b*
Bodové ohodnocení je orientační, mohlo se lišit podle závažnosti chyby (někdo danou vlastnost neřešil vůbec, jiný se snažil a něco mu uteklo).

Zbývajících 5 bodů bylo možno získat za formálně správný zápis ZA, tedy za sedmici a zápis přechodové funkce. Opravdu jsem se divila, kolik různých zápisů jste vynalezli. Ráda bych zde alespoň zopakovala, že když se něco jmenuje funkce, zapisuje se to jako funkční_symbol(argumenty) = výsledek; v případě přechodové funkce ZA je výsledkem množina, takže je třeba použít množinové závorky.

Jako správné jsem uznávala i rozšířené ZA (je ale slušností uvést, že víte, že používáte rozšířený automat, a že víte, že je ekvivalentní základnímu ZA). Bylo tedy možné řešit příklad tak, že pro jazyk navrhnete gramatiku a tu pak analyzujete. Někteří tento přístup použili, mnoho gramatik však bylo špatně. Za správnou gramatiku bez automatu bylo možno získat max. 20 bodů.

Podzim 2009, 25.1.2010, Termín 03

Příklad 2
35 bodů

Uvažme následující pravou kongruenci $\sim$ na slovech nad abecedou $\Sigma = \{a, b\}$:

$$u \sim v \quad \stackrel{def}{\Longleftrightarrow} \quad \#_a(u) \bmod 4 \;=\; \#_a(v) \bmod 4$$

(a) Určete index $\sim$.

(b) Najděte jazyk L nad Σ takový, že $\sim_L = \sim$.

(c) Najděte jazyk L, který je sjednocením některých tříd rozkladu Σ^* podle $\sim$, ale přitom $\sim_L \neq \sim$.

ad a)
Reseni:
index ~ je roven 4

Za spravne reseni jste ziskali 5 bodu.
Za ostatni reseni jste ziskali 0 bodu.
(0 bodu jste ziskali i v pripade, ze jste uvedli, ze index ~ je roven 5, jelikoz trida slov neobsahujici zadna 'a' neodpovida tride slov, kde pocet 'a' modulo 4 je roven 0 - to neni samozrejme pravda)

ad b)
Reseni:
Napriklad jazyk L = {w \in {a,b}^* | #_a(w) mod 4 = 0}

Za spravne reseni jste ziskali 15 bodu.
Za nespravne reseni jste ziskali 0 bodu.
Pokud Vase reseni bylo pouze automat (gramatika) akceptujici (generujici) spravny jazyk ziskali jste 10 bodu.
Nekteri z Vas zapomneli na pismeno 'b' a psali reseni jako napriklad L = {a^i | i mod 4 = 0}. Za podobne reseni jste ziskali 5 bodu.
Za vyslovene spatny zapis spravneho jazyka (typicky spatne zavorky) jsem strhnul 1 bod.

ad c)
Reseni:
Napriklad jazyk L = {w \in {a,b}^* | #_a(w) mod 2 = 0}

Za spravne reseni jste ziskali 15 bodu.
Za nespravne reseni jste ziskali 0 bodu.
Pokud Vase reseni bylo pouze automat (gramatika) akceptujici (generujici) spravny jazyk ziskali jste 10 bodu.
Nekteri z Vas zapomneli na pismeno 'b' a psali reseni jako napriklad L = {a^i | i mod 2 = 0}. Za podobne reseni jste ziskali 5 bodu.
Za vyslovene spatny zapis spravneho jazyka (typicky spatne zavorky) jsem strhnul 1 bod.

U vetsiny nespravnych reseni, ktere davaly aspon nejaky smysl, jsem se snazil ukazat, proc Vas jazyk nesplnuje zadani.

To znamena:
ad b) jsem typicky uvadel slova u,v takova, ze u ~ v, ale neplati, ze u ~L v
ad c) jsem typicky uvadel slova u,v takova, ze u ~ v, ale zaroven u \in L ale neplati, ze v \in L.

Podzim 2009, 25.1.2010, Termín 03

Je dána gramatika $\mathcal{G} = (\{S, A, B\}, \{a, b, c\}, P, S)$, kde

Příklad 3
35 bodů

$$P = \{ \begin{array}{lcl} S & \rightarrow & AaA \mid ab \mid B \\ A & \rightarrow & BaBb \mid c \mid B \\ B & \rightarrow & Bc \mid S \ \}. \end{array}$$

Převeďte gramatiku $\mathcal{G}$ na ekvivalentní gramatiku v CNF. Pokud nepoužijete algoritmus z přednášky, zdůvodněte ekvivalenci výsledné gramatiky s gramatikou ze zadání.

Hodnotenie a časté chyby:
Algoritmus na prevod gramatiky na prevod do CNF nájdete v skriptách. Keďže má tento algoritmus ako vstup vlastnú gramatiku, bolo predtým treba odstrániť jednoduché pravidlá. Najviac ste za tretí list mohli dostať 35 bodov, ktoré boli rozdelené nasledovne (i s častými penalizáciami):

```
  15 bodov - odstránenie jednoduchých pravidiel
    -8 bodov za vynechanie pravidiel A -> ab, A-> AaB
  15 bodov - prevod vlastnej gramatiky do CNF
    -2,3 body za každú drobnejšiu chybu
    -6 a viac za všetky výskyty systematickej chyby
  5  bodov - výsledok, ktorým je gramatika v CNF
    -2 body za vynechanie niektorých neterminálov
    -3 body ak nebola k pravidlám vôbec napísaná štvorica
    -4 body ak gramatika nebola v CNF
```

Záporné body sa neprenášali medzi časťami.
Každá chyba sa hodnotila len na mieste, kde sa vyskytla, teda ak bol zlý výsledok odstránenia jednoduchých pravidiel, i tak ste mohli získať za zvyšné časti 20 bodov. Jediná výnimka sú -4 body za to, že výsledok nebol v CNF, ktoré sa strhávali vždy. Tu som bol prísny, pretože overenie je čisto syntaktické a mohli ste si to sami veľmi jednoducho skontrolovať.
Viacerí z vás sa snažili o odstránenie pravej rekurzie. Okrem štandardnej penalizácie za zbytočnosť vo forme typicky 2 bodov ste sa pripravili i o ďalšie body, pretože sa myslím nikomu nepodarilo previesť to bez chýb.

Podzim 2009, 25.1.2010, Termín 03

Zformulujte algoritmus, který k danému nedeterministickému konečnému automatu $\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$ zkonstruuje jazykově ekvivalentní deterministický automat bez nedosažitelných stavů a s totální přechodovou funkcí. *(Nezapomeňte přesně popsat výstupní automat.)*

Příklad 4
40 bodů

Hodnocení:

Alespoň nějaký trochu slušný pokus o algoritmus, i když nedělal, to co měl, jsem hodnotil 5 body. To se týkalo zejména těch, kteří zapomněli na determinizaci a napsali algorimus pro odstranění nedosažitelných stavů a "ztotálnění" přechodové funkce nedeterministického automatu.

U velké části z Vás bylo vidět, že jste schopni provést determinizaci na konkrétním příkladě, ale napsat obecný algoritmus je už problém. Proto jste pouze neformálně popsali, jak by se ta determinizace asi dělala. Řešení tohoto typu jsem hodnotil 10-25 body, podle toho, jak dobře by se podle uvedeného neformálního popisu dalo postupovat. Nic přesnějšího k hodnocení těchto řešení tady nejsem schopen říci, protože v podstatě každé bylo originální.

Pro získání více jak 25 bodů bylo potřeba (se aspoň slušně pokusit) uvést přesný postup, jak se vybuduje množina dosažitelných stavů výsledného automatu, a přesně definovat přechodovou funkci výsledného deterministického automatu. To jest, čemu se přesně rovná \delta'(S, a), kde S je množina stavů původního automatu a "a" je symbol abecedy, nikoliv uvést něco jako - "Ze stavu S se můžu dostat tam, kam se mohl původní automat dostat ze všech stavů z S.". Podle toho, jak dobře se Vám to povedlo jste získali 30, 35, nebo 40 bodů.

Podzim 2009, 25.1.2010, Termín 03

Rozhodněte, zda existují následující gramatiky. V kladném případě uveďte příklad takové gramatiky, v záporném důkaz její neexistence.

Příklad 5
15+15 bodů

(a) Bezkontextová gramatika bez jednoduchých pravidel a bez ε-pravidel, která je cyklická.

(b) Bezkontextová gramatika, která má vlastnost sebevložení, ale generuje regulární jazyk.

Řešení:
(a) NE, neexistuje. Důkaz: Nechť G = (N,E,P,S) je bezkontextová gramatika bez jednoduchých a \eps-pravidel. Pak všechna pravidla jsou jednoho z tvarů:
* A -> a, kde A \in N, a \in E
* A -> \alfa, kde A \in N, |\alfa| >= 2
* S -> \eps, kde S se nevyskytuje na pravé straně žádného pravidla

Pak kdykoli A =>+ \beta, tak \beta \in (E \cup {\eps}) nebo |\beta| >= 2, a proto je vyloučeno \beta = A (=cyklus).
(b) ANO, existuje. Např. G = ({S},{a},{S -> aSa|a},S).

Hodnocení:
** 5 bodů za správnou odpověď ano/ne, 10 za důkaz (a) / gramatiku (b)*
** 1-3 body při chybné odpovědi ano/ne, ale prokázání relevantní znalosti*
** (a) přidělení poměrné části z 10ti bodů v případě důkazu, který dostatečně nevysvětloval důvod, proč zadaná gramatika neexistuje*

Podzim 2009, 25.1.2010, Termín 03

Příklad 6
35+5+5 bodů

(a) Definujte *Turingův stroj* (včetně podmínky kladené na přechodovou funkci).

(b) Jak se nazývá třída jazyků akceptovaných Turingovými stroji?

(c) Jak se nazývá třída jazyků akceptovaných úplnými Turingovými stroji?

(a) Definujte Turingův stroj (včetně podmínky kladené na přechodovou funkci).

Řešení + bodování:
Turingův stroj je devítice $(Q, \Sigma, \Gamma, \triangleright, \sqcup, \delta, q_0, q_{accept}, q_{reject})$[3b], kde:

- Q je konečná [1,5b] množina stavů [2b],
- Σje konečná [1,5b] množina vstupních symbolů [2b],
- Γje konečná [1,5b] množina páskových symbolů [2b], $\Sigma \subseteq \Gamma$[2b],
- $\triangleright \in \Gamma \setminus \Sigma$[1b] je levá koncová značka [2b],
- $\sqcup \in \Gamma \setminus \Sigma$[1b] je symbol prázdného políčka [2b],
- $\delta : (Q \setminus \{q_{accept}, q_{reject}\}) \times \Gamma \to Q \times \Gamma \times \{L, R\}$[4b] je totální [0,5b] přechodová funkce [1,5b],
- $q_0 \in Q$je počáteční stav [2b],
- $q_{accept} \in Q$je akceptující stav [2b],
- $q_{reject} \in Q$je zamítající stav [2b], a
- přechodová funkce nikdy nepřepíše levou koncovou značku a neposune se po čtení levé koncové značky doleva, tj. $(\forall q \in Q)(\exists p \in Q)(\delta(q, \triangleright) = (p, \triangleright, R))$[1,5b].

Celkem tedy 35 bodů. Kromě toho jsem jako ~~bonus~~ malus strhával 4 body, kdykoli se na papíře objevilo slovo "zásobník" (+ jeho odvozeniny, jako "zásobníková" apod.) nebo tvrzení: "Turingův stroj je PDA, který má navíc..."
Několik málo lidí místo formální definice psalo slohové práce na téma Turingův stroj, ty jsem hodnotil maximálně 10 body. (Tím neříkám, že se ta definice nedala napsat slovně. Dala, ale nikomu, kdo se o to pokusil, se to ani zdaleka nepodařilo.)

(b) Jak se nazývá třída jazyků akceptovaných Turingovými stroji?

Řešení: Třída rekurzivně spočetných jazyků. [5b]

(c) Jak se nazývá třída jazyků akceptovaných úplnými Turingovými stroji?

Řešení: Třída rekurzivních jazyků. [5b]

Bonus – výběr z nesprávných a nesmyslných odpovědí (pro b i c): regulární, bezkontextové, kontextové, DCFL, frázové, TML, Turingové, konečné, spočetné, konečně spočetné, úplné, naddimenzionální (WTF?), konfigurácia (cože?), všechny jazyky. (Poslední možná stojí za zamyšlení – proč neakceptují Turingovy stroje všechny jazyky?)

Podzim 2009, 3.2.2010, Termín 04

O každém z následujících jazyků rozhodněte, zda je bezkontextový. Svá tvrzení dokažte. *(Pro důkaz, že jazyk je bezkontextový, stačí napsat odpovídající gramatiku nebo automat.)*

Příklad 1
50 bodů

(a) $L_1 = \{w \in \{a,b,c\}^* \mid \#_a(w) \leq \#_c(w) \ \wedge \ \#_b(w) \leq \#_c(w)\}$

(b) $L_2 = \{w \in \{a,b,c\}^* \mid \#_a(w) = 2 \cdot \#_b(w)\}$

Ad a) Jazyk neni bezkontextovy. Pro dukaz teto skutecnosti jste mohli vyuzit P.L. pro bezkontextove jazyky. Mohli jste ziskat az 25 bodu. Priklad spravneho pouziti P.L.

Necht n \in N je konstanta z P.L.
Volime slovo z = a^nb^nc^n
Nyni uvazime vsechna rozdeleni slova z na podslova u,v,w,x,y takova, ze
|vx|>0 a |vwx|<= n a pro kazde toto rozdeleni ukazeme ze existuje i \in N_0
takove, ze slovo z' = ux^iwy^iz nepatri do jazyka L1.
Uvazme rozdeleni, kde podslovo vwx obsahuje pouze znak(y):

'a'- volime i = 2 a dostavame, ze
ux^iwy^iy = z' \in {a,b,c}^*, kde #a(z') > #c(z')
'b'- volime i = 2 a dostavame, ze
ux^iwy^iy = z' \in {a,b,c}^*, kde #b(z') > #c(z')
'a,b'- volime i = 2 a dostavame, ze
ux^iwy^iy = z' \in {a,b,c}^*, kde #a(z') > #c(z') a #b(z') > #c(z')
'b,c'- volime i = 0 a dostavame, ze
ux^iwy^iy = z' \in {a,b,c}^*, kde #a(z') > #c(z')
'c'- volime i = 0 a dostavame, ze
ux^iwy^iy = z' \in {a,b,c}^*, kde #a(z') > #c(z') a #b(z') > #c(z')

Jak je videt, pro kazde pripustne rozdeleni jsme ukazali, ze existuje i \in N_0, takove, ze slovo z' = ux^iwy^iz nepatri do jazyka L1 a tudiz z P.L. pro bezkontextove jazyky plyne, ze jazyl L1 neni bezkontextovy.

Typicke chyby.
Nepochopeni P.L. nebo jeho spatne pouziti. Typicky jste nezvolili slovo, zvolene slovo nesplnovalo podminky P.L. nebo jste zvolili slovo, ktere obsahovala nejake nezname. V tomto pripade jste typicky ziskalit 7 bodu.
Pokud jste zvolili slovo, pro ktere existovalo spatne rozdeleni, maximalne jste ziskali 15 bodu.
Dalsi chybou bylo neuvazeni vsech platnych rozdeleni nebo jejich nepresne popsani. Typicky jsem srazel 3 az 10 bodu dle zavaznosti
V tomto prikladu bylo dulezite zvolit vhodny zpusob jak popsat vsechna pripustna rozdeleni zvoleneho slova z na podslova u,v,w,x,y a jak o nich ukazat, ze existuje i \in N_0, takove ze slovo ux^iwy^iz nepatri do jazyka L1.
Ruzne zvolene zpusoby pak implikovaly ruznou obtiznost potrebne argumentace. Nekteri z Vas zvolili zpusoby, ktere nebyly zcela presne a bylo jim strzeno 3 az 10 bodu dle zavaznosti.
Pokud jste odpovedeli pouze ne bez dukazu dostali jste 5 bodu.

Ad b) Jazyk je bezkontextovy. Pro dukaz teto skutecnosti stacilo napsat odpovidajici gramatiku nebo zasobnikovy automat. Mohli jste ziskat az 25 bodu.

Priklad spravne gramatiky:
G=({S},{a,b,c},P,S)
P = { S -> SaSaSbS | SaSbSaS | SbSaSaS | cS | epsilon }

Typicke chyby.
Formalni nedostatky jako napr. chybejici definice (hlavicka) gramatiky nebo automatu, chybejici zpusob akceptovani, nepresne definovana prechodova funkce a podobne nepresnosti. Za tyto chyby Vam bylo strzeno 0 az 3 body dle zavaznosti.
Dalsi chyby byly v nepresnem navrhu gramatiky ci automatu jako napr:
- opomenuti, ze epsilon \in L2 - srazka 2 body
- uvazovali jste jazyk L2 = {w \in {a,b} | #b(w) = 2*#a(w) } - srazka 5 bodu*
- chyby souvisejici s vyskytem symbolu 'c' - srazka 5 bodu
- automat (gramatika) nedovoloval akceptovat (generovat) vsechna slova z jazyka - srazka 5 - 10 bodu
- automat (gramatika) dovoloval akceptovat (generovat) slova se spatnym poctem 'a' a 'b' - srazka 18 bodu
Pokud jste odpovedeli pouze ano bez uvedeni automatu ci gramatiky dostali jste 5 bodu.

Podzim 2009, 3.2.2010, Termín 04

K zadanému konečnému automatu zkonstruujte ekvivalentní deterministický konečný automat s totální přechodovou funkcí.
(Pokud nepoužijete standardní algoritmus, dokažte ekvivalenci obou automatů.)

Příklad 2
30 bodů

	c	d
$\rightarrow 1$	$\{5\}$	$\{4\}$
2	$\{4\}$	$\{2,3\}$
$\leftarrow 3$	$\{1\}$	$\{1,3\}$
$\leftarrow 4$	$\emptyset$	$\{1\}$
5	$\{3,5\}$	$\{1,5\}$
$\leftarrow 6$	$\{2,3\}$	$\{1,4,6\}$

odstráním nedosažitelné stavy a z totálním automat

⇒

	c	d
→ 1	{5}	{4}
← 3	{1}	{1,3}
← 4	{N}	{1}
5	{3,5}	{1,5}
N	{N}	{N}

⇒

25

⇒

	c	d
→ {1}	{5}	{4}
{5}	{3,5}	{1,5}
← {4}	N	{1}
← {3,5}	{1,3,5}	{1,3,5}
~~←~~ {1,5}	{3,5}	{1,4,5}
N	N	N
← {1,3,5}	{1,3,5}	{1,3,4,5}
← {1,4,5}	{3,N,5}	{1,4,5}
← {3,N,5}	{N,1,3,5}	{N,1,3,5}
← {N,1,3,5}	{N,1,3,5}	{N,1,3,5}

Hodnotenie a časté chyby:
Algoritmus na transformáciu NFA na DFA nájdete v skriptách. Ak ste postupovali podľa neho a došli k správnemu výsledku, mohli ste získať 30 bodov.

Za správnu inicializáciu algoritmu bolo 5 bodov. Myslené je tým to, že ste začali počiatočným stavom, nie napr. množinou všetkých stavov zadaného automatu, čo viedlo okrem iného k množstvu nedosiahnuteľných stavov. Ďalších 10 bodov bolo za postup, ktorý viedol k ekvivalentnému automatu, 4 body k tomu, ak bol skutočne deterministický (je potrebné rozlišovať stavy a množiny stavov). Ďalej 5 bodov, ak bola prechodová funkcia naozaj totálna (bolo možné ľahko overiť) a 6 bodov za správne označenie počiatočného a všetkých koncových stavov. Ak ste označili viacero počiatočných stavov, strhol som za to 4 body (bolo zrejmé, že sa jedná o chybu, keďže DFA má už z definície len 1 iniciálny stav).

Často ste automat zo zadania pred použitím algoritmu pozmenili, pridali ďalší stav ("čiernu dieru") a presmerovali do neho prechod zo stavu 4 pod "c". Tento krok bol zbytočný, keďže štandardný algoritmus môžme aplikovať na ľubovoľný NFA a výsledkom je DFA s totálnou prechodovou funkciou, viedol k zväčšeniu výsledného automatu, ale nebolo to brané ako chyba. Ak ste použili štandardný algoritmus, takisto nebolo nutné na začiatku explicitne odstraňovať nedosiahnuteľné stavy. Napriek tomu, ak ste tieto stavy odstránili a neurobili v príklade nič iné, bral som to tak, že ak by ste mali viac času, možno to mohlo k niečomu zmysluplnému viesť a symbolicky som to ocenil. Na konci ste sa často pokúšali o minimalizáciu, čo bolo takisto zbytočné a navyše ste tým väčšinou zaviedli do riešenia ďalšie chyby, čím ste prišli o 2 a viac bodov (podľa počtu chýb). Drobné chyby (chybné spočítanie nejakého stavu, zabudnutie na nejaký stav, formálne nedostatky a pod.) boli za -1/-2 body.

Implicitne som predpokladal, že ste sa rozhodli použiť štandardný algoritmus a snažil som sa jeho myšlienku aspoň sčasti nájsť aj vo veľmi kreatívnych postupoch. V prípade, že váš postup algoritmus z prednášky ani zďaleka nepripomínal, bola nastavená stupnica na 20 bodov za samotné riešenie a 10 bodov za dôkaz ekvivalencie automatov (neobjavil sa).

Podzim 2009, 3.2.2010, Termín 04

Je dána gramatika $\mathcal{G} = (\{S, A, B\}, \{a, b\}, P, S)$, kde

Příklad 3
20+15 bodů

$$
\begin{aligned}
P = \{\ & S \rightarrow aAA \mid Bb \\
& A \rightarrow aB \mid bb \\
& B \rightarrow Bb \mid bS \mid \varepsilon\ \}.
\end{aligned}
$$

33

(a) Zkonstruujte PDA $\mathcal{A}$ pro nedeterministickou syntaktickou analýzu shora dolů. Uveďte způsob akceptování.

(b) Zapište akceptující výpočet automatu $\mathcal{A}$ nad slovem $aabbb$.

a) $\mathcal{A} = (\{q\}, \{a,b\}, \{S, A, B, a, b\}, \delta, q, S, \emptyset)$, kde δ ...

akceptuje prázdným zásobníkem

$\delta(q, \varepsilon, S) = \{(q, aAA), (q, Bb)\}$

$\delta(q, \varepsilon, A) = \{(q, aB), (q, bb)\}$

$\delta(q, \varepsilon, B) = \{(q, Bb), (q, bS), (q, \varepsilon)\}$

$\delta(q, a, a) = \{(q, \varepsilon)\}$

$\delta(q, b, b) = \{(q, \varepsilon)\}$

20

b) $q, (aabb, S) \vdash^{\varepsilon} (aabb, aAA) \vdash^{a} (abb, AA) \vdash^{\varepsilon} (abb, aBA) \vdash^{a} (bb, BA) \vdash^{\varepsilon}$
$\vdash (bb, A) \vdash^{\varepsilon} (bb, bb) \vdash^{b} \vdash^{b} (\varepsilon, \varepsilon)$

$\Rightarrow$ ~~automat~~ PDA akceptuje slovo aabb

Špatné slovo Vám odpouštím

13

Několik málo poznámek k bodování:
Za vytvoření ZA provádějícího analýzu jste mohli získat 20 bodů, z toho 5 připadlo na formální zápis sedmice (např. správný výčet symbolů zásobníkové abecedy se leckomu nezdařil, kupodivu jste si také pletli stavy automatu a symboly zásobníkové abecedy) a přechodové funkce (opět jsem žasla nad všelijakými zápisy :( ) Zbylých 15 bodů bylo rovnoměrně rozděleno za zapsání jednotlivých řádků přechodové funkce.

V druhé části příkladu jste mohli získat 15 bodů za předvedení výpočtu Vašeho automatu nad daným slovem. Zde jsem požadovala správný zápis konfigurace včetně stavu, jeho neuvedení jsem většinou hodnotila -2 body. Za drobné chyby v odvozování (např. použití pravidla, které v gramatice nebylo, či přehlédnutí nějakého symbolu v zásobníku) jsem strhávala 3 body. Největší chybou bylo použití přepisovacího pravidla na symbol, který nebyl na vrcholu zásobníku, to jsem hodnotila většinou -10 body (v případech, kdy se toto stalo jen v jednom případě, což bylo možno pokládat za omyl, byla srážka nižší).

Chci upozornit, ze jelikoz je nedeterministicky PDA pro analyzu shora dolu pouze jednostavovy, prijali jsme konvenci, ze v zapisu vypoctu tohoto automatu lze vynechat stav ve vsech konfiguracich. Stejna konvence je zminena i ve skriptech (str. 85). Strhnuti dvou bodu za vyuziti teto konvence je tedy nespravne.

Mate-li tyto dva body strzeny a ma-li to vliv na hodnoceni predmetu, prihlaste se mi prosim mailem a ja to napravim.

Podzim 2009, 3.2.2010, Termín 04

Příklad 4
40 bodů

Zformulujte algoritmus, který k dané bezkontextové gramatice $\mathcal{G} = (N, \Sigma, P, S)$ bez ε-pravidel zkonstruuje jazykově ekvivalentní gramatiku bez jednoduchých pravidel a bez ε-pravidel. *(Nezapomeňte přesně popsat výstupní gramatiku.)*

Řešení je ve skriptech, takže ho sem psát nebudu.

Hodnocení:

Za ukázku na příkladu bylo 5 bodů.

Za obecné popsání odstranění jednoduchého pravidla pomocí substituce bylo 10 bodů.

Za neurčitý a neformální popis správného algoritmu pro odstraňování jednoduchých pravidel bylo 5-25 bodů, podle toho, jak dobře by se podle uvedeného neformálního popisu dalo postupovat. Nic přesnějšího k hodnocení těchto řešení tady nejsem schopen říci, protože v podstatě každé bylo originální.

Vyskytlo se celkem hodně řešení, která byla sice dobře zapsána, ale byla chybná. Nejčastější byly asi varianty následujícího algoritmu.

Dokud gramatika obsahuje nějaké jednoduché pravidlo X -> Y, odstraň toto pravidlo pomocí substituce za Y.
I když se toto řešení obohatí o zahazování pravidel typu X -> X, existují gramatiky, na kterých tento algoritmus cyklí. Nedá se mu ale upřít jistá myšlenka, a tak jsem ho hodnotil 15 body.

15 body bylo hodnoceno i řešení, které pouze provedlo substituce za existující jednoduchá pravidla a ignorovalo možnost řetězů jednoduchých pravidel (např. A -> B, B -> C, C -> abc).

Pro získání více jak 25 bodů bylo potřeba (se aspoň slušně pokusit) uvést přesný postup, jak se vybuduje množina N_A pro každý neterminál A \in N a jak se vytvořené množiny použijí k odstranění jednoduchých pravidel. Podle toho, jak dobře se Vám to povedlo jste získali 30, 35, nebo 40 bodů.

Vyskytla se dvě správné řešení, která nepoužívala algoritmus ze skript, i ty samozřejmě byly hodnoceny 40 body.

Podzim 2009, 3.2.2010, Termín 04

Rozhodněte, zda existují následující gramatiky. V kladném případě uveďte příklad takové gramatiky, v záporném důkaz její neexistence.

Příklad 5
15+15 bodů

(a) Bezkontextová gramatika bez jednoduchých pravidel, která je cyklická.

$\emptyset$

(b) Frázová gramatika, která není kontextová, ale generuje konečný jazyk.

Zadání: Rozhodněte, zda existují následující gramatiky a uveďte příklad nebo důkaz neexistence.

(a) Bezkontextová gramatika bez jednoduchých pravidel, která je cyklická.

Řešení: ANO, např. G = ({S}, {a}, {S -> SS | \epsilon}, S).

(b) Frázová gramatika, která není kontextová, ale generuje konečný jazyk.

Řešení: ANO, např. G = ({S, A}, {a}, P, S), kde
P = { S -> AA
AA -> a }
Tato gramatika generuje jazyk {a}.

Jiné řešení (párkrát se něco podobného vyskytlo, ačkoliv jsem si téměř jist, že omylem a že to rozhodně nebyl váš původní úmysl):
G = ({S}, {a}, P, S), kde
P = {aS -> a}
Tato gramatika generuje prázdný jazyk(!).

Hodnocení: Odpověď ANO/NE (nebo aspoň něco, co odpověď naznačuje, jako třeba uvedení aspoň nějaké gramatiky) - 5b, gramatika - 10b.

Typické srážky: -3b, pokud u části (b) chyběla čtveřice G=(...), z podobných důvodů, jaké minule psala kolegyně Bühnová (spousta těch, co čtveřici psala, ji napsala špatně); -1b za různé formální nepřesnosti; zbylé srážky byly specifické a jsou popsány přímo na listu.

Podzim 2009, 3.2.2010, Termín 04

Příklad 6
25+15 bodů

(a) Zformulujte Myhill-Nerodovu větu.

(b) Nechť $\mathcal{M} = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ je PDA. Napište podmínky, které musí platit, aby byl zásobníkový automat *deterministický*.

11

a) Formulujte Myhillovu-Nerodovu větu. (25 bodů) Příklad správné odpovědi:

Nechť L je jazyk nad abecedou Σ, pak následující tvrzení jsou ekvivalentní (5 bodů, stačilo mi i napsat, že M-N věta dává ekvivalenci tří tvrzení):

- L je rozpoznatelný konečným automatem (popř. L je regulární) (5 bodů)
- L je sjednocením některých tříd rozkladu (množiny Σ^*) určeného pravou kongruencí na Σ^* s konečným indexem (8 bodů)
- index prefixové ekvivalence pro jazyk L je konečný (7 bodů)

Pokud jste u druhého tvrzení nenapsali, že relace určující daný rozklad musí mít konečný index, strhával jsem tři body. Podobně pokud jste neuvedli, že má jít o pravou kongruenci. Pokud jste napsali, že L je sjednocením některých tříd rozkladu určeného prefixovou ekvivalencí, dostali jste pouze tři body, neboť toto tvrzení platí vždy, bez ohledu na to, zda je L regulární (podobně pro jiné tautologie týkající se prefixové ekvivalence). Třetí tvrzení jste většinou uvedli správně. Pokud jste větu formulovali ve tvaru "L je regulární, pak (právě když) jsou splněny následující tři podmínky", strhával jsem dva body, neboť Vaše tvrzení je sice pravdivé, avšak M-N věta je silnější (říká navíc, že pokud L není regulární, tak žádná ze zmíněných podmínek neplatí, zatímco Vaše tvrzení říká, že alespoň jedna podmínka neplatí).

Pár řešitelů formulovalo alternativní variantu M-N věty uvedenou ve skriptech ("Počet stavů minimálního automatu je roven indexu prefixové ekvivalence...") Toto řešení jsem uznal za správné, pouze pokud bylo výslovně uvedeno, že tento minimální automat existuje právě tehdy, když je index prefixové ekvivalence konečný (ve skriptech to uvedeno je). Jinak jsem strhával podstatnou část bodů.

b) Měli jste uvést podmínky, které musí splňovat zásobníkový automat, aby byl deterministický. Správná odpověď viz skripta či slidy. Nahlédnutím do skript zjistíte, že podmínky byly dvě, za každou jsem dával 7,5 bodů (půlbody jsem zaokrouhloval nahoru). Drtivá většina z Vás zapomínala na podmínku týkající se \epsilon-kroků, takže jsem za ni dával poměrně hodně bodů i v tom případě, že nebyla zformulována formálně zcela korektně. U druhé podmínky byl největším problémem její formální zápis (resp. jeho nedostatek), za který jsem strhával 2-5 bodů.

Jak tedy snadno nahlédnete, hodnocení bylo velmi mírné, navíc se po Vás nechtěl v podstatě žádný myšlenkový výkon. Bohužel jsem byl celkovým výsledkem tohoto příkladu zklamán, zejména v první části jsem se namísto jasně strukturovaného tvrzení musel probírat změtí slov a symbolů, které často nedávaly smysl (občas ani po gramatické stránce). Vždy jsem ale hledal něco, co by alespoň trochu dávalo část tvrzení M-N věty, abych mohl udělit částečné body. A to i přes to, že jsem mnohdy pochyboval o tom, zda vůbec chápete takové základní pojmy, jako je relace ekvivalence, pravá kongruence, rozklad množiny apod. Vybrané perly:
-"relace vlnka je sjednocením tříd rozkladu podle relace vlnka"
-"jazyk má konečný počet tříd rozkladu"
-"každé slovo padne do konečně mnoha tříd" (aby taky ne, když vždy patří právě do jedné)
-...a konečně moje oblíbené: "M-N věta tvrdí, že každý jazyk L lze rozpoznat konečným automatem" (nebo něco na ten způsob). To jsou ovšem pánové Myhill a Nerode větší borci, než Chuck Norris.

Podzim 2010 , 7.1.2011, Termín 01

Navrhněte zásobníkový automat $\mathcal{M}$ akceptující jazyk

Příklad 1
40 bodů

$$L = \{a^i b^j c^k \mid i, j, k > 0,\ i \neq j\ \vee\ i \neq k\}.$$

Uveďte, jakým způsobem navržený automat akceptuje.

Jedno z možných vzorových řešení najdete na http://fi.muni.cz/~xkunc7/ag/automat.jpg Přidal jsem pro lepší pochopení i popis toho, jak automat pracuje. Kdyby to náhodou nešlo po mně přečíst můžu to vysázet do pdf. Automat není "nejkratší ani nejhezčí", ale mělo by z něj být jasně vidět, jak postupuje. Pomocí stavů si hlídá, které znaky abecedy smí načítat a pomocí znaků na zásobníku hlídá počty.

HODNOCENÍ PŘÍKLADU:
Celkově se dělil příklad na dvě skupiny řešitelů.
První část přišla na princip nedeterminismu a ta druhá ne.

Všichni z druhé skupiny se většinou marně snažili pomocí zásobníku napočítat áčka, poté odmazat béčka a nějak znovu napočítat céčka. To ovšem není možné už kvůli prvnímu příkladu Pumping lemmy pro CFL, kdy bylo ukázáno, že a^n.b^n.c^n, n \in N není CFL, protože nedokážeme počítat tři znaky "zvlášť". Tato skupina byla odsouzena k záhubě od začátku kvůli deterministickému postupu, kdy po odmazání znaků ze zásobníku pro porovnání a <> b, již nemohu napočítat c, protože nevím, kolik bylo a.
Tato skupina se velmi těžko hodnotila, ale většinou jste dostali 1 - 15 bodů (nejčastěji kolem osmi), podle toho, zda jste aspoň správně napsali sedmici vašeho automatu, jaký jazyk ve skutečnosti automat akceptoval, zda dodržoval správné pořadí znaků abc, zda aspoň zazněla myšlenka počítat "odděleně", i když jste nakonec počítali dohromady a některé další aspekty. Často byl výsledkem automat, který akceptoval #a(w) <> #b(w) + #c(w) nebo jste splnili aspoň první podmínku #a(w) <> #b(w) a mnohé další řešení v podsatě jste to měli téměř každý jinak (nedobře).

Druhá skupina správně odhalila myšlenku nedeterminismu (ať už stavem nebo symbolem na zásobníku) a zapracovala jej do automatu. Za samotnou myšlenku bylo 5 bodů, podle správnosti jejího využití (rozhodnout se včas, nepřepínat mezi dvěma průchody, umožnit aspoň jeden znak a, b i c atd.) až 20 bodů.
Za každou část automatu i <> j a i <> k jste mohli získat dalších 10 bodů, tady bylo hodnoceno, zda část automatu opravdu akceptuje to, co tvrdíte, dodržuje pořadí znaků a-b-c (častý problém), vynucuje aspoň jeden znak od každého, správnost způsobu akceptování (občas byl automat správně, pokud by akceptoval jinak než jste tvrdili) a další drobnosti.

Body jste mohli dále ztratit:
Rozšířený zásobníkový automat (v zadání nebylo explicitně zmíněno, stejně jako u domácího úkolu!):
Pokud jste jednou využili nenačítání žádného symbolu z vrcholu -2 b.
Pokud jste tohoto využili víckrát -5 b.
Pokud byl celý automat postaven na načítání více symbolů z vrcholu, pak bylo maximum 30 bodů (hlavně kvůli chybám, který automat většinou pak obsahoval).
Pokud jste automat popsali pouze slovně nebo obrázkem, maximum bylo za zcela intuitivní a dobrý popis 25 bodů -> tímto popisem nedosáhnete vhodného formalismu a není zcela jasné, zda jste ošetřili veškeré potřebné podmínky.
Až -5 bodů za chybějící sedmici či chyby v ní (to se nedělá jen tak: bylo by dobré uvést, kterým stavem začínám, co mám na zásobníku a JAK akceptuji).
Za drobné chyby a chyby v zápisu jsem body nestrhával (max. 1 bod celkem).

Několik z vás sestrojilo gramatiku a poté pro ni syntaktický analyzátor shora dolů.
OK, ale gramatika se povedla asi jen dvěma z vás. Zbytku byly body sraženy podle podobných podmínek jako výše.

Ještě nakonec: další body byly strženy, pokud jste pouze tvrdili, že automat sestrojím jako sjednocení dvou automatů (a nebylo řečeno jak), případně tvrdili, že sestrojím paralelní synchronní kompozici automatů. Tento pojem je přesně definován (skripta Definice 2.9) jen pro FA s totálními přech. funkcemi. U zásobníkových nic takového neznáte (zkuste si rozmyslet proč).

Na úplný závěr mého (opět) dlouhého příspěvku si ještě neodpustím: většina z vás neumí pracovat s přechodem pod epsilon (ať už načítání žádného znaku abecedy nebo žádného symbolu ze zásobníku (u rozšířeného)). Většinou s tím pracujete jakože už nemám co číst (dočtené slovo, nebo prázdný zásobník), ale neuvědomujete si sílu tohoto výrazu. Například delta(q, eps, eps) = {něco}, znamená, že kdykoli jsem ve stavu q, pak mohu tento přechod provést, ne jen tehdy, když "se mi to hodí"!

Podzim 2010 , 7.1.2011, Termín 01

Příklad 01

$$\left(\{q_0, q_1, q_{1F}, q_2, q_3, q_{2F}\}, \{a,b,c\}, \{A,=,Z\}, \delta, q_0, Z, \underbrace{\{q_{F_1}, q_{F_2}\}}_{\text{AKCEPTUJI STAVEM}}\right) = M$$

$\delta(q_0, a, Z) = \{(q_0, A{=}Z)\}$ → NA ZAČÁTKU SI VLOŽÍM NAD DNO $\underline{Z}$ ZNAK =, KTERÝ INDIKUJE ROVNOST

$\delta(q_0, a, A) = \{(q_0, AA)\}$ → KOLIK $\underline{a}$, TOLIK $\underline{A}$ na zásobník

! $\delta(q_0, b, A) = \{(q_1, \varepsilon), (q_2, A)\}$ - POKUD MÁM ASPOŇ 1 $\underline{A}$ A NAČÍTÁM $\underline{b}$ - ROZHODNI SE, ZDA POČÍTÁME $\underline{b}$ NEBO $\underline{c}$

1: $i \neq j$ 2: $i \neq k$

1.1 $\delta(q_1, b, A) = \{(q_1, \varepsilon)\}$

1.2 $\delta(q_1, b, =) = \{(q_1, \varepsilon)\}$

1.3 $\delta(q_1, b, Z) = \{(q_1, Z)\}$

1.4 $\delta(q_1, c, Z) = \delta(q_1, c, A) = \{(q_{1F}, A)\}$

1.F $\delta(q_{1F}, c, A) = \{(q_{1F}, A)\}$

2.1 $\delta(q_2, b, A) = \{(q_2, A)\}$

2.2 $\delta(q_2, c, A) = \{(q_3, \varepsilon)\}$

2.3 $\delta(q_3, c, A) = \{(q_3, \varepsilon)\} = \delta(q_3, c, =)$

2.4 $\delta(q_3, c, Z) = \{(q_3, Z)\}$

2.5 $\delta(q_3, \varepsilon, Z) = \delta(q_3, \varepsilon, A) = \{(q_{2F}, \varepsilon)\}$

$(L = \{a^i b^j c^k \mid i,j,k > 0,\ i \neq j \lor i \neq k\})$

1+2: NEJPOZDĚJI V TOMTO BODĚ SE MUSÍM ROZMYSLET, ZDA POČÍTÁM $\underline{b}$ NEBO $\underline{c}$. DOČETL JSEM $\underline{a}$ A PŘESNĚ TAKOVÝ POČET $\underline{A}$ MÁM NA ZÁSOBNÍKU, POD NIMI SYMBOL $\underline{=}$ A POD NÍM DNO Z.

1: POČÍTÁM $\underline{b}$, PROTO HNED ZA PRVNÍ $\underline{b}$ STRHÁVÁM $\underline{A}$

1.1: ZA KAŽDÉ $\underline{b}$ STRHNI $\underline{A}$

1.2: POKUD PŘIJDE $\underline{b}$ A MÁM $\underline{=}$ NA VRCHOLU, VÍM, ŽE JSEM PRÁVĚ DOSTAL O 1 ZNAK $\underline{b}$ VÍCE NEŽ $\underline{a}$

1.3 TÍM PÁDEM JSEM ODHALIL DNO A MOHU PŘIDÁVAT LIBOVOLNĚ $\underline{b}$

1.4 PŘECHOD DO AKCEPTUJÍCÍHO STAVU JE PODMÍNĚN ASPOŇ 1 ZNAKEM $\underline{c}$. ZÁROVEŇ MOHU AKCEPTOVAT, POKUD $i \neq j$, TEDY MÁM NA VRCHOLU $\mathbf{A}$ ($\#a(w) > \#b(w)$) NEBO $\underline{Z}$ ($\#b(w) > \#a(w)$). VE STAVU 1.F PŘIDÁM LIBOVOLNĚ $\underline{c}$

2.1 $\underline{b}$ MĚ NEZAJÍMAJÍ - ZÁS. NEMĚNÍM

2.2 DOST $\underline{b}$ → PRVNÍ $\underline{c}$ → ODMAŽU $\underline{A}$ → DALŠÍ STAV, KDE MOHU POUZE $\underline{c}$

2.3 ZA KAŽDÉ $\underline{c}$ UMAŽU $\underline{A}$, PŘÍPADNĚ V MOMENTĚ $i = k$ SYMBOL $\underline{=}$

2.4 PO = VÍM, ŽE $\#c(w) > \#a(w)$, PŘIDÁVÁM LIBOVOLNĚ DALŠÍ $\underline{c}$

2.5 POKUD NENÍ NA VRCHOLU = MOHU AKCEPTOVAT

Podzim 2010 , 7.1.2011, Termín 01

Uvažme jazyk $L = \{a\}.(\{a,b\}.\{a,b\})^*$ nad abecedou $\Sigma = \{a,b\}$.

Příklad 2
17+8+8 bodů

(a) Určete index relace $\sim_L$ a popište jednotlivé třídy ekvivalence $\sim_L$.

(b) Najděte jazyk L' nad Σ takový, že $L' \neq L$ a $\sim_{L'} = \sim_L$.

26

(c) Najděte jazyk L'', který je sjednocením některých tříd rozkladu Σ^* podle $\sim_L$, ale přitom $\sim_{L''} \neq \sim_L$.

ad a) index ~L = 4 (minimalni deterministicky automat ma 4 stavy) jednotlive tridy ekvivalence jsou:
- T_0 = {epsilon}
- T_1 = L
- T_2 = L.{a,b}
- T_3 = {b}.{a,b}*

Celkove jste mohli ziskat 17 bodu:
- 5 bodu za index
*- 3 body za kazdou spravnou tridu (4 * 3 = 12)*

Caste chyby:
Vas automat nebyl totalni (chybel stav pro "peklo" - trida T_3)
*- zisk 11 bodu (2 body za index ~_L = 3 a 3 * 3 bodu za 3 tridy ekvivalence)*
Vas automat nebyl minimalni:
- zisk 3 bodu za kazdou spravnou tridu ekvivalence (typicky 9 bodu)
Nepresny zapis jednotlivych trid ekvivalence
- srazka 0 - 3 body dle zavaznosti

ad b) Zde existovalo vicero spravnych jazyku, napriklad:
L' = co-L, L' = T_2, L' = T_0 U T_1 atd.

Za spravny jazyk jste ziskali 9 bodu.

Caste chyby:
L' sice byl roven sjednoceni nekterych trid rozkladu {a,b}^* podle ~_L, ale neplatilo, ze ~_L' = ~_L (spravne reseni pro pripad c))
- zisk 2 bodu

Dale jste casto uvadeli jazyky, ktere nejsou ani sjednocenim nekterych trid rozkladu {a,b}^* podle ~_L (napriklad L' = {b}.({a,b}.{a,b})^*)
- zisk 0 bodu

Nepresny zapis jazyka
- srazka 0 - 2 body dle zavaznosti

ad c) Zde existovalo rovnez vicero spravnych jazyku, napriklad:
L'' = T_0, L'' = T_3, L'' = T_1 U T_2 atd.

Za spravny jazyk jste ziskali 9 bodu.

Caste chyby:
L'' sice byl roven sjednoceni nekterych trid rozkladu {a,b}^* podle ~_L, ale neplatilo, ze ~_L'' != ~_L (spravne reseni pro pripad b))
- zisk 2 bodu

Dale jste casto uvadeli jazyky, ktere nejsou ani sjednocenim nekterych trid rozkladu {a,b}^* podle ~_L (napriklad L'' = {a} nebo L'' = ({a,b}.{a,b})^*)
- zisk 0 bodu

Nepresny zapis jazyka
- srazka 0 - 2 body dle zavaznosti

Podzim 2010 , 7.1.2011, Termín 01

Je dána gramatika $\mathcal{G} = (\{S, A, B\}, \{a, b\}, P, S)$, kde

Příklad 3
20+15 bodů

$$
\begin{aligned}
P = \{\ S &\to AaB \mid Bb \\
A &\to aB \mid aa \\
B &\to Bb \mid bA \mid \varepsilon\ \}.
\end{aligned}
$$

31

(a) Zkonstruujte rozšířený PDA $\mathcal{A}$ pro nedeterministickou syntaktickou analýzu zdola nahoru. Uveďte způsob akceptování.

(b) Zapište akceptující výpočet automatu $\mathcal{A}$ nad slovem $aaba$.

a) akceptuje koncovým stavem

$\mathcal{A} = (\{q_0, q_f\}, \{a, b\}, \{a, b, S, A, B, \bot\}, \delta, q_0, \bot, \{q_f\})$ kde δ je dále definována

$\delta(q_0, \varepsilon, AaB) = \{(q_0, S)\}$ −1

$\delta(q_0, \varepsilon, Bb) = \{(q_0, S)\}$

$\delta(q_0, \varepsilon, aB) = (q_0, A)$

$\delta(q_0, \varepsilon, aa) = (q_0, A)$

−2 $\delta(q_0, \varepsilon, Bb) = (q_0, B)$

$\delta(q_0, \varepsilon, bA) = (q_0, B)$

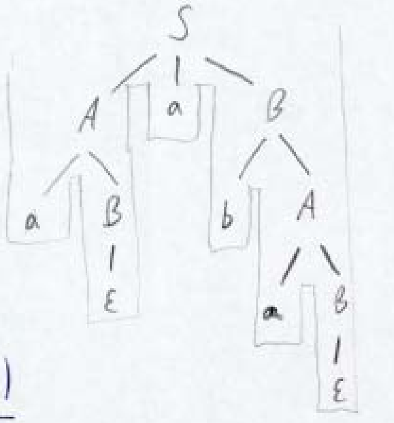
$\delta(q_0, \varepsilon, \varepsilon) = (q_0, \varepsilon)$

$\delta(q_0, a, \varepsilon) = (q_0, a)$

$\delta(q_0, b, \varepsilon) = (q_0, b)$

$\delta(q_0, \varepsilon, \bot S) = (q_f, \varepsilon)$

Derivační strom pro slovo aaba



b)

−1

$(q_0, aaba, \bot) \vdash^{a} (q_0, aba, \bot a) \vdash^{\varepsilon} (q_0, aba, \bot aB) \vdash^{\varepsilon} (q_0, aba, \bot A)$

$\vdash^{a} (q_0, ba, \bot Aa) \vdash^{ba} (q_0, \varepsilon, \bot Aaba) \vdash^{\varepsilon} (q_0, \varepsilon, \bot AabaB)$

$\vdash^{\varepsilon} (q_0, \varepsilon, \bot AabA) \vdash^{\varepsilon} (q_0, \varepsilon, \bot AaB) \vdash^{\varepsilon} (q_0, \varepsilon, \bot S) \vdash^{\varepsilon} (q_f, \varepsilon)$

$\Rightarrow$ slovo $\boxed{aaba}$ je akceptováno PDA $\mathcal{A}$

Vzhledem k tomu, že se jednalo o jednoduchý příklad, kdy stačilo držet se vzoru ze skript/cvičení, hodnotila jsem spíš přísně (-1 bod za formální chyby jako závorky, definice prvků sedmice apod., -2 body za chybu v přechodové funkci, obzvlášť hrozné chyby -víc). Je škoda, že jste kolikrát ztráceli body za věci, které jste si mohli vyzkoušet v domácím úkolu a na které opravující ve svém komentáři ještě výslovně upozorňoval - typicky dvojí definice přechodu d(q,e,bB)={(q,B)}, d(q,e,bB)={(q,S)} místo d(q,e,bB)={(q,B),(q,S)}. Příklad také ukázal, že řada z Vás moc nechápe, jak vlastně zásobníkový automat funguje - co asi může akceptovat automat, který nikdy nečte ze vstupu? A hříšníci, co ve výpočtu automatu měli přechody se změnami uprostřed zásobníku, ať si na wiki najdou pojem zásobník.

Podzim 2010 , 7.1.2011, Termín 01

Nechť L je regulární jazyk přijímaný nedeterministickým konečným automatem $\mathcal{M} = (Q, \Sigma, \delta, q_0, F)$. Popište konstrukci, kterou z automatu $\mathcal{M}$ získáme konečný automat akceptující jazyk

Příklad 4
40 bodů

$$L' = \{w \in \Sigma^* \mid a.w \in L\}.$$

Tento priklad lze resit nejmene tremi ruznymi postupy:

1. Puvodni automat zdeterminizuji a pak prohlasim za inicialni ten stav, kam se z puvodniho inicialniho stavu dostanu pod a. Pokud prechod z puvodniho inicialniho stavu pod a neexistuje, vratim automat akceptujici prazdny jazyk.

2. Pridam novy stav, ze ktereho povedou epsilon-prechody do vsech stavu, kam vedou z puvodniho inicialniho stavu pod a. Nove pridany stav prohlasim za inicialni.

3. Pridam novy stav q0', ktery bude inicialni. Pro kazde dva stavy q,q' a symbol b takove, ze v puvodnim stavu existuji prechody q0 -a-> q -b-> q' (kde q0 je puvodni inicialni stav), pridam do noveho automatu prechod q0' -b-> q'.

Jak vidite, kazde z tech tri reseni se vejde na tri radky. V zadani nebylo zamerne presne receno, jakeho typu ma byt vysledny automat. Taky tam bylo napsano, ze vstupni automat je NFA, nikoliv NFA s epsilon kroky.

Bodování a časté chyby:

Správných řešení bylo poměrně dost, za zcela správné řešení bylo samozřejmě 40 bodů. Pokud jste správně uvedli myšlenku jednoho ze 4 výše uvedených postupů, ale pokulhával zápis, dostali jste alespoň 30 bodů (dle toho, jak moc velké chyby jste dělali... za drobné formální chybky jsem strhával řádově jednotky bodů, za vyloženě zmatený zápis pak víc). 35 bodů dostal ten, kdo dal návod na sestrojení automatu pro jazyk {w|aw \in L, a \in Σ} - v zadání nebyla u 'a' žádná kvantifikace, tím pádem se tím myslí jeden konkrétní symbol (a i těm, kteří se u zkoušky ptali, jsem říkal, že 'a' je pochopitelně symbol abecedy, ne že je to libovolný symbol abecedy).

Bodové rozpětí 10-25 bodů bylo určeno těm, kteří směle vykročili ke správnému řešení, ale v některém zásadním aspektu bylo to řešení špatně. Často jste si neuvědomili, že do stavu q_0 mohou rovněž vést přechody a některé přechody vedoucí z tohoto stavu (či stav samotný) jste odstranili (čímž se pochopitelně zásadně mění akceptovaný jazyk, pokud skutečně existuje cyklus vedoucí přes q_0). Například 10 bodů jsem dával, pokud jste pouze smazali všechny přechody vedoucí z q_0 pod jiným symbolem než 'a'. 20 bodů jsem dával, pokud jste apriori předpokládali, že M je DFA (tzn. vynechali jste ve variantě 1 determinizaci). Vyšší počet bodů byl za řešení, kde se s cykly přes q_0 nějakým způsobem počítalo, avšak řešily se špatným způsobem.

Pokud někdo pouze popsal algoritmus pro determinizaci, dával jsem maximálně 10 bodů. Stejný počet bodů dostal ten kdo konstruoval automat pro jazyk {aw|w \in L}. Maximálně 5 bodů pak dostali ti, kteří konstruovali automat akceptující jazyk {a}.Σ

Body jsem též strhával za zmatený zápis a chyby ve formálním zápisu. Bohužel se opět objevilo nemálo řešení, z nichž se vůbec nedal dekódovat úmysl autora. Proto si dovolím malý tip k přípravě na další písemky: pokud se učíte/konzultujete ve skupinkách, využijte toho a dejte si navzájem přečíst svá řešení příkladů, ve kterých je zapotřebí přesným způsobem prezentovat nějaký postup. Pokud váš sparing-partner není schopen toto řešení dekódovat, zkuste jej ještě jednou pořádně projít a zapsat lépe. Opakujte až do doby, dokud nebude příslušný text pochopitelný. K této činnosti můžete využít i diskusní fórum (například můžete vkládat vlastní naskenovaná řešení).

Rovněž bych chtěl upozornit že 'formální zápis' není totéž, co zápis pomocí spousty matematických značek. Byla řešení, která se skládala ze tří vět v českém/slovenském jazyce a jednoho či dvou matematických symbolů a přesto za ně byl plný počet bodů, neboť zmíněný text přesně popisoval, co se má se vstupním automatem udělat. Naopak některé stránky plné změti symbolů dávaly jasně najevo, že autor jejich význam vůbec nechápe. Mějte tedy na paměti, že při popisu nějakého postupu má nejvyšší prioritu přesnost a jednoznačnost tohoto popisu (když tedy nepočítám jeho správnost, samozřejmě:)).

Podzim 2010 , 7.1.2011, Termín 01

Rozhodněte, zda existují následující gramatiky. V kladném případě uveďte příklad takové gramatiky, v záporném důkaz její neexistence.

Příklad 5
15+15 bodů

(a) **Bezkontextová gramatika, která má vlastnost sebevložení a generuje jazyk, který nemá vlastnost sebevložení.**

(b) **Bezkontextová gramatika v CNF, která je cyklická.**

a) Správné řešení: ANO, existuje, například: G = ({S},{a},{S -> aS | Sa | a},S) generuje jazyk L = {a}^+.

Hodnocení: za pouhou odpověď ANO - 2 body, za odpověď ANO s nějakým textem okolo, který ale nebyl příklad gramatiky, jen pokus o "zdůvodnění" toho, že taková gramatika může existovat - 5 bodů. Za špatný příklad gramatiky (nemá vl. sebevložení nebo jazyk, který generuje, není regulární) - 5 bodů. Za správný příklad gramatiky - 15 bodů. Bodové srážky (à -1 až -3 body) za formální nedostatky, špatný zápis, nesmyslná tvrzení apod

b) Správné řešení: NE, taková gramatika neexistuje.
Zdůvodnění: Gramatika v CNF má pravidla těchto tří tvarů: A -> BC (kde A, B, C jsou neterminály), A -> a (kde A je neterminál, a je terminál) a S -> epsilon (kde S je počáteční neterminál nevyskytující se na žádné pravé straně pravidla). Gramatika je cyklická, pokud je v ní možné odvození A =>^+ A pro nějaký neterminál A. Zřejmě při takovém odvození nemůže být použito pravidlo S -> epsilon, je totiž povolenou pouze pokud S není na pravé straně, mohlo by tedy být použito pouze v prvním kroku. Rovněž není nikdy v takovém odvození možné použít pravidla tvaru A -> a, protože by se tím do větné formy dostal terminál. Zbývají tedy pouze pravidla tvaru A -> BC, ale ta vždy prodlužují větnou formu o jeden symbol, jejich použitím tedy není možné v nenulovém počtu kroků dostat z větné formy délky jedna opět větnou formu délky jedna. Cyklická gramatika v CNF tedy nemůže existovat.

Poznámka: Takto podrobné řešení jen píšu pro vaši představu, jak správně vypadá důkaz. V podstatě mi z tohoto důkazu stačilo uvést to, že pravidla typu A -> a do větné formy přidávají terminál, pravidla typu A -> BC větnou formu prodlužují a pravidlo S -> epsilon je možno použít jen na začátku. Takováto forma důkazu byla ale mezi vašimi písemkami poměrně hodně vzácná. Akceptoval jsem proto i jako správný argument tvrzení "gramatika v CNF je bez jednoduchých a epsilon pravidel, proto nemůže být cyklická".

Hodnocení: 2 body za prosté NE, 5 - 15 bodů za NE plus zdůvodnění, podle kvality zdůvodnění a počtu nesmyslů v něm obsažených.

Časté nesmysly:
- gramatika v CNF je vlastní (to není pravda)
- vstup do CNF je vlastní gramatika (to je nesmysl, jaký "vstup do CNF"?)
- při převodu na CNF nejprve odstraňujeme epsilon pravidla a jednoduchá pravidla (to je špatný argument, to že na vstupu algoritmu pro CNF je gramatika nějakého tvaru, nemusí vůbec souviset s tím, jakého tvaru jsou gramatiky v CNF; spousta z vás si plete pojem CNF jakožto vlastnost gramatiky s algoritmem na převod do CNF, jde o naprosto různé věci)
- gramatika v CNF nemá jednoduchá pravidla, proto nemůže být cyklická (to opět není pravda)

Podzim 2010 , 7.1.2011, Termín 01

Nechť $\mathcal{M} = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ je PDA.

Příklad 6
8+6+9+14+8 bodů

(a) Napište typ funkce δ. 7

(b) Definujte *konfiguraci* PDA. 6

(c) Definujte relaci *krok výpočtu* ($\vdash_{\mathcal{M}}$). 0

(d) Napište podmínky, které musí PDA splňovat, aby byl *deterministický*. 11

(e) Rozhodněte, zda je třída DCFL uzavřená na následující operace.
(Správnou odpověď zakroužkujte.) 4

průnik:	JE	NENÍ
sjednocení:	JE	NENÍ
doplněk:	JE	NENÍ
průnik s reg. jazykem:	JE	NENÍ

ad a) Napiště typ funkce δ z definice PDA. Správná odpověď zní $Q\times(\Sigma \cup\{\varepsilon\})\times\Gamma\rightarrow P\_fin(Q\times\Gamma^*)$, kde P_fin značí množinu všech konečných podmnožin. Za příklad jste mohli získat až 8 bodů. 4 body za typ vstupu, 4 typ hodnot. Nejčastější chyby a bodové srážky:

- *chybějící ε na levé straně: -1b*
- *přebívající iterace na levé straně: -1b*
- *chybějící P_fin: -2b, pokud jste alespoň uvedli, že výsledkem funkce je množina, tedy 2^(Q×Γ*), strhával jsem 1b. Pokud jste slovně okometovali, že se jedná o množinu konečných podmnožin, body jsem nestrhával.*
- *chybějící * na pravé straně: -1b.*
- *mnozí z vás uvedli příklad, místo typu. Body jsem uděloval v závislosti na dobrém komentáři.*

ad b) Definujte konfiguraci PDA. Konfigurace je libovolná trojice (p, w, α) z $Q\times\Sigma^*\times\Gamma^*$, kde p značí aktuální stav automatu, w nepřečtenou část vstupního slova a α značí aktuální obsah zásobníku. Za příklad bylo možné dostat 6 bodů. Nejčastěji jste zapomínali, že w a α jsou slova nad svými abecedami, ne znaky(3b).

ad c) Definujte relaci krok výpočtu. Binární relace na množině všech konfigurací je definována takto: $(p,aw,Z\alpha) \vdash (q,w,\gamma\alpha) \Leftrightarrow \exists(q,\gamma) \in \delta(p,a,Z)$ pro $a \in\Sigma\cup\{\varepsilon\}$, $\alpha,\varphi \in\Gamma^*$, $w\in\Sigma^*$. Za správnou definici jste mohli získat 9 bodů.

- *Pokudy o slovní popis dopadly většinou nevalně. U definice je potřeba vyjadřovat se přesně. Většinou max 3 body.*
- *Absence domén použitých symbolů. Spolu s drobnými nepřesnostmi či přepisy většinou -1 bod.*
- *Formulace typu krok výpočtu udává přechod automatu apod. jsem honoroval 0b.*

ad d) Popsat podmínky , které musí splňovat PDA, aby byl deterministický. Podmínky jsou 2. První říká, že kdykoliv je δ(q,ε,Z), pro nějaký stav q a vrchol zásobníku Z, neprázdná, tak pro všechny a zΣ platí, že δ(q,a,Z) je prázdná. Druhá podmínka zní: Pro libovolné a z množiny $\Sigma \cup\{\varepsilon\}$, libovolný stav q a libovolný vrchol zásobníku Z platí: $|\delta(q,a,Z)| \le 1$.
Příklad byl oceněn 14 body, každá podmínka za 7. Nedopadl tak špatně, jako předchozí dva.

- *Drobné nepřesnosti, podle závažnosti 0-3 body na každou z podmínek.*
- *Opomenutí ε u druhé podmínky -1b.*
- *Při naznačení alespoň základní myšlenky jste mohli získat 1b za každou.*
- *Některé slovní popisy byly velmi zdařilé, pokud byly přesné, byly oceněny plným počtem bodů.*

ad e) Čtyři otázky JE - NENÍ, hodnoceno mírně - body za špatné odpovědi jsem nestrhával. Správné řešení:NENÍ,NENÍ,JE,JE. 2 body za každou.

*Obecné komentáře: Valná většina z vás má velký problém s přesným vyjadřováním. Pokud máte něco definovat, definujte přesně, nepřesná definice je špatná definice. Definice není popis toho, jak se daná věc používá, ale co definovaná věc je. Vždy se nad vámi psanou definicí zamyslete a představte si, že byste ji vyčetli ze skript. Pokud vám taková představa evokuje nadávku na učitele typu "Co to ***** je?", "To nedává přece smysl...", "Vždyť to neříká vůbec nic..." tak je něco v nepořádku. Formulace typu "⊦ čteme z pásky symbol M a přecházíme do jiného stavu" sice pobaví kdekoho, nicméně to neříká o daném nic jiného, než že by si měli dívky dávat pozor. Neuvěřitelné množství z vás místo definice konfigurace PDA popisovalo význam symbolů použitých v definici automatu. Zajímavá epidemie...*

Podzim 2010 , 13.1.2011, Termín 02, Příklad 01

Vasim ukolem v 1.listu bylo pro nasledujici jazyky rozhodnout, zda jsou bezkontextove.
a) L1 = {a^ib^jc^k | i,j,k >= 0, j>=2i}
b) L2 = {a^ib^jc^k | i,j,k >= 0, j=2i a k=2j}

Ad a) Jazyk je bezkontextovy. Pro dukaz teto skutecnosti stacilo napsat odpovidajici gramatiku nebo zasobnikovy automat. Mohli jste ziskat az 25 bodu.
Priklad spravne gramatiky:
G=({S,A,B,C},{a,b,c},P,S)
P = { S -> AC | A | C | epsilon
    A -> abb | aAbb | B
    B -> bB | b
    C -> cC | c }

Typicke chyby.
Formalni nedostatky jako napr. chybejici definice (hlavicka) gramatiky nebo automatu, chybejici zpusob akceptovani, nepresne definovana prechodova funkce a podobne nepresnosti. Za tyto chyby Vam bylo strzeno 0 az 3 body dle zavaznosti.

Dalsi chyby byly v nepresnem navrhu gramatiky ci automatu jako napr:
- nezachovani poctu jednotlivych pismen - srazka 10 az 15 bodu
- nezachovani poradi jednotlivych pismen - srazka 6 az 10 bodu
- generovani (akceptovani) pouze slov, kde j=2i - srazka 6 az 8 bodu
- generovani (akceptovani) pouze slov, kde je sudy pocet 'b' - srazka 6 az 8 bodu
- chyby v "okrajovych" pripadech
  - generovani (akceptovani) pouze slov kde j>2i - srazka 2 body
  - negenerovani (neakceptovani) slov b^+, c^+, b^+c^+ - srazka 2 - 4 body

Pokud jste odpovedeli pouze ano bez uvedeni automatu ci gramatiky dostali jste 2 body.

Ad b) Jazyk neni bezkontextovy. Pro dukaz teto skutecnosti jste mohli vyuzit P.L. pro bezkontextove jazyky. Mohli jste ziskat az 25 bodu.
Priklad spravneho pouziti P.L.
Necht n \in N je konstanta z P.L.
Volime slovo z = a^nb^{2n}c^{4n}
Nyni uvazime vsechna rozdeleni slova z na podslova u,v,w,x,y takova, ze
|vx|>0 a |vwx|<= n a pro kazde toto rozdeleni ukazeme ze existuje i \in N_0
takove, ze slovo z' = uv^iwx^iy nepatri do jazyka L2.
Uvazme rozdeleni, kde podslovo vwx obsahuje pouze znak(y):

'a'- volime i = 0 a dostavame, ze
  uv^iwx^iy = a^{n-k}b^{2n}c^{4n} pro 0 < k <= n
'b'- volime i = 0 a dostavame, ze
  uv^iwx^iy = a^{n}b^{2n-k}c^{4n} pro 0 < k <= n
'c'- volime i = 0 a dostavame, ze
  uv^iwx^iy = a^{n}b^{2n}c^{4n-k} pro 0 < k <= n
'a,b'- volime i = 0 a dostavame, ze
  uv^iwx^iy = a^{n-k}b^{2n-l}c^{4n} pro k,l > 0 a 0 < k+l <= n
'b,c'- volime i = 0 a dostavame, ze
  uv^iwx^iy = a^{n}b^{2n-k}c^{4n-l} pro k,l > 0 a 0 < k+l <= n

Jak je videt, pro kazde pripustne rozdeleni jsme ukazali, ze existuje i \in N_0,
takove, ze slovo z' = ux^iwy^iz nepatri do jazyka L2 a tudiz z P.L. pro
bezkontextove jazyky plyne, ze jazyl L2 neni bezkontextovy.

Typicke chyby.
Nepochopeni P.L. nebo jeho spatne pouziti. Typicky jste nezvolili slovo nebo zvolene slovo nesplnovalo podminky P.L. (napr z= a^nb^nc^n). V tomto pripade jste ziskali maximalne 5 bodu. Pokud jste uvazili jen nekolik malo konkretnich rozdeleni, mohli jste ziskat maximalne 10 bodu
Dalsi chybou bylo neuvazeni vsech platnych rozdeleni nebo jejich nepresne popsani. Typicky jsem srazel 3 az 10 bodu dle zavaznosti.
V tomto prikladu bylo dulezite zvolit vhodny zpusob jak popsat vsechna pripustna rozdeleni zvoleneho slova z na podslova u,v,w,x,y a jak o nich ukazat, ze existuje i \in N_0, takove ze slovo ux^iwy^iz nepatri do jazyka L2.
Ruzne zvolene zpusoby pak implikovaly ruznou obtiznost potrebne argumentace. Nekteri z Vas zvolili zpusoby, ktere nebyly zcela presne a bylo jim strzeno 3 az 10 bodu dle zavaznosti. Za formalni nedostatky v dukaze jsem strhaval 0 az 5 bodu dle zavaznosti
Pokud jste odpovedeli pouze ne bez dukazu dostali jste 2 body.

Podzim 2010 , 13.1.2011, Termín 02, Příklad 02

regulární výrazy jsou už též opravené a čekají vedle skeneru, až na ně přijde řada. Zdá se, že tento příklad byl poměrně obtížný - snažila jsem se nehodnotit moc přísně, ale i tak to nedopadlo nejlíp.

Správných řešení existuje více, nejjednodušší byla asi tato:

a) L1 = {ab,a} - {a,b}*.{aaa}.{a,b}*

RE: (a.b + a.a.b)*.(e + a + a.a)

b) L2 = {w nad {a,b} | délka w je sudá a počet 'a' ve w je alespoň 1}

RE: (b.b)* . (a.a + a.b + b.a) . (a.a + a.b + b.a + b.b)*

K výsledku se dalo dopracovat buď metodou "kouknu a vidím", nebo pomocí konečného automatu a jeho algoritmického převodu na RE. První přístup je rychlejší, ale náchylnější k chybám, u druhého se aspoň dají dávat body za postup. Kromě správnosti řešení, příp. postupu jsem hodnotila i formální správnost výsledného výrazu srážkami 1 bod za chybu (typicky "pozitivní iterace" - ta u reg. výrazů není definovaná, sjednocení místo + apod.). Někteří z Vás se pokusili o řešení pomocí rozdílů, průniků a doplňků RE, to neprošlo.

Časté chyby:

** Zapomenuté/špatně ohodnocené přechody KA při odstraňování uzlů - mnoho chyb z nepozornosti, některým ale není moc jasné fungování cyklů.*

** Chybný KA - je dobré si vyzkoušet, jestli tím, co jste vytvořili, projde pár základních slov jazyka - typicky je dobré zkusit ta nejkratší, potom nějaké delší.*

** Chybný RE navržený bez KA - opět, zkuste si, jestli tím projde pár základních slov z jazyka! Často v případě a) chyběla slova a, aa*

** (ab)*.(ba)*.(aa)*.(bb)* není totéž co (ab+ba+aa+bb)**

** (a.(ab)*)* popisuje mimo jiné i slova typu a^n*

Podzim 2010 , 13.1.2011, Termín 02, Příklad 03

příklad 3 (analýza shora dolů) dopadl velmi dobře. Většina z Vás v tom má jasno, ale bohužel se často objevovaly formální nedostatky. Nejčastěji () vs. {}, = vs. ->, -> vs. krok výpočtu, {prazdna mnozina} vs. prazdna mnozina, atd. Nezapomínejte, že součástí automatu je také sedmice a součástí jeho zásobníkové abecedy všechny terminály i neterminály. Bodově byl příklad hodnocen podobně jako minule, tedy za formální nedostatky po bodu až dvou dolů.

Podzim 2010 , 13.1.2011, Termín 02, Příklad 04

Vaším úkolem **bylo zformulovat algoritmus pro odstranění jednoduchých pravidel z gramatiky, která neobsahuje epsilon pravidla a uvést, co je výsledná gramatika**. Správné řešení můžete nalézt ve slajdech k 8. přednášce, nebo, trochu složitější verzi, ve skriptech. Obě uvedená řešení jsou psaná v pseodokódě, nicméně mi stačil srozumitelný a správný slovní popis.

Obě uvedená správná řešení se skládají ze dvou částí. Napočítání pomocných množin N_A a pak popisem, jaká pravidla zařadit do nově konstruované množiny pravidel. Každou část jsem honoroval nejvýše 18 body(pokud jste se snažili o takto dělené řešení). Zbývající 4 body jsem uděloval za správný zápis finální gramatiky. Na to jste často zapomínali i přesto, že to bylo výslovně uvedeno vzadaní.

Napočítání pomocných množin:
*Zde šlo o to, napočítat si množiny N_A={B \in N| A=>*B} pro každé A z množiny neterminálů. Pokud jste uvedli alespoň takovýhle zápis(nebo jiný, dále užitečný), obdrželi jste max 8 bodů. Pokud byl uveden i(nebo pouze) postup, mohli jste obdržet až zmíněných 18 bodů. Bodů za velké nesmysly jsem strhával 3-5, za velké přehmaty i více a za malé nepřesnosti většinou 1-3 body dohromady. Bodoval jsem až plným počtem bodů i jiné množiny, které vám mohli dále nějak pomoci. např. s množinami N_A'={B \in N| B=>+A} se dá dopracovat k rozumnému a fungujícímu algoritmu.*

Přidání pravidel do nové P':
Myšlenka byla většinou dobrá, popis provedení ovšem horší. Někteří z vás dostali zde velmi málo bodů, ikdyž jsem po chvíli luštění, sebezapření a přistoupení na váš slovník pochopil, že algoritmus ovládáte. A to proto, že vaším úkolem bylo zformulovat algoritmus a ne popsat, jak algoritmus používáte. Navíc některé vaše formulace neznamenají, co si myslíte, že znamenají.

Pokud jste si napočítali jiné množiny, než ty, které by se vám hodily v druhé fázi algoritmu, strhával jsem 10 bodů.

Velmi často se objevovalo "naivní" řešení, které jen nahrazovalo jednoduchá pravidla A->B pravidly, A->beta_1,..,beta_n kde B->beta_1,..,beta_n \in P. Jinými slovy nahrazením pravé strany jednoduchého pravidla všemi pravými stranami pravidel pro B. Za toto a jemu podobná řešení jsem uděloval maximálně 15 bodů. Příklad pravidel gramatiky, pro kterou takový algoritmus nezastaví, či ponechá v gramatice jednoduchá pravidla je:
P''={S->a|A,
A->b|B,
B->c|A}

Nejčastější chyby a nesmysly:
** Do pravidel z A přidej všechna nejednoduchá pravidla z N_A. - Ať tento příkaz aplikuji libovolně dlouho, nebudu mít nikdy co přidat, protože N_A obsahuje pouze neterminály. To je ale mé jediné štěstí, protože bych zase nevěděl, kam je přidat, když A je neterminál a ne množina.*
**A->*B - tohle jsme si nikdy nedefinovali.*
** Věty typu "neterminálu přidáme pravidla neterminálu, do kterého jsme se dostali přes jednoduchá pravidla." - To zavání trochu automaty a opět s těmi pravidly neterminálu... Neterminál nemá pravidla.*
**Místo čtveřice přesně popisující výstupní gramatiku jste často uváděli jen "gramatika bez epsilon a jednoduchých pravidel." Ale co jste zamýšleli jako výstup algoritmu, zůstalo utajeno.*
**Používání N jako symbolu pro jeden neterminál. (např. S=>N) správný zápis: S=>A, A \in N. Na to pozor!*
**Algoritmus pro odstranění epsilon-pravidel nebyl třeba - vstup byl bez epsilon pravidel.*
**slovo pravidlo se skloňuje podle vzoru město, ne žena! (pravidly, ne pravidlami)*

Teď snad jen pár slov a rad závěrem. Rada číslo jedna - nebojte se používat zástupné symboly. Popisovat souvětím na 5 řádek, že množinou N_i máte na mysli N_A, je zbytečné. Nebo: Pravidlo A->B nahradím A->alfa, kde B->alfa \in P je daleko srozumitelnější (a dle mého jednoduší napsat), než když se to samé snažíte zapsat bez A,B,alfa. Jednodušeji se pracuje s konkrétními symboly než s abstraktními pojmy. Pokud nevíte přesně, jak různá značení používat (a na cviku tedy nevíte, co cvičící píše), obraťte se na něj s dotazem, může vám to velmi usnadnit studium. A než nějaké značení zavedete, věnujte mu třebas 10 vteřin na rozmyšlenou, jestli je dané označení vhodné a jednoduše použitelné. A teď jedna prosba. Pokud píšete sloh na dvě strany A4, snažte se vyjadřovat přesně, jednoduše a používejte vaši mateřštinu správně. Souvětím přes celou stránku se také zuste vyhnout. Není totiž vždy jednoduché je správně rozkódovat.

Podzim 2010 , 13.1.2011, Termín 02, Příklad 05

Zadání: Rozhodněte, zda existují následující jazyky. Pokud ano, uveďte příklad. Pokud ne, dokažte.

(a) Bezkontextový jazyk L, který není regulární a splňuje L = L^2.

Správné řešení: ANO, např. L = { w \in {a,b}^* | #a(w) = #b(w) }

Hodnocení: 2 body za pouhou odpověď ANO, 5 - 15 bodů za (pokus o) příklad jazyka, splňujícího zadání, případné bodové srážky za nesmyslná tvrzení, chyby v zápisu (kolikrát potřebujete číst, že L = a^nb^n není zápis jazyka, abyste to pochopili?) apod.

Typické chyby:
- jazyk, který splňoval dvě ze tří podmínek v zadání, hodnocen 7b
- jazyk, který splňoval jen jednu ze tří podmínek v zadání, hodnocen 5b

(b) Bezkontextový jazyk L, který není regulární a přitom L^2 je konečný.

Správné řešení. NE, kdyby byl L^2 konečný, pak by nutně musel být i L konečný (druhá mocnina nekonečného jazyka je zřejmě opět nekonečná).

Hodnocení: 2 body za pouhé NE, 5 - 15 bodů za (pokus o) důkaz, bodové srážky za nesmyslná tvrzení apod.

Typické chybné pokusy o důkaz:
- "Je-li L^2 konečný, pak je regulární, regulární jazyky jsou uzavřené na mocninu, a proto je i L regulární."
Nesmysl, protože tvrzení "L^2 regulární => L regulární" neplatí. Hodnotil jsem 7 body (minus případné srážky za další nesmysly).

- "Bezkontextové jazyky nejsou konečné a jsou uzavřené na zřetězení. Proto je-li L bezkontextový, je i L^2 bezkontextový a tím pádem není konečný."
Nesmysl, kde jste přišli na to, že bezkontextový jazyk nemůže být konečný? Hodnotil jsem 5 body.

- "Bezkontextové jazyky jsou uzavřené na zřetězení, a proto je-li L^2 konečný, musí být i L konečný."
Správný závěr, ale nesmysl na začátku věty. Tvrzení o konečnosti L vůbec nijak nesouvisí s uzavřeností bezkontextových jazyků na cokoli. Strhával jsem za podobné výlevy 3 body.

Poznámka na závěr: Zapamatujte si, prosím už projednou, že tvrzení "třída jazyků X je uzavřena na operaci OP" znamená:
Jsou-li L1, L2 jazyky ze třídy X, pak L1 OP L2 je jazyk ve třídě X.

Rozhodně to NEznamená:
**Je-li L1 OP L2 ve třídě X, pak i L1, L2 jsou ve třídě X.*

Podzim 2010 , 13.1.2011, Termín 02, Příklad 06

a) Definujte, kdy je bezkontextová gramatika necyklická (10 bodů)

Správná odpověď: viz slajdy k 8. přednášce (slajd 10), či skripta (strana 71).

Bodování a časté chyby:

Velmi častou chybou byl zmatený zápis, kdy jste podmínku z definice psali ve tvaru "neexistuje pravidlo A=>^+ A" nebo dokonce "neexistuje pravidlo A->^+ A" (symbol ->^+ nebyl nikde definován). Pokud jste v podmínce psali o neexistenci pravidla a zároveň použili symbol -> dostali jste maximálně 5 bodů. Pokud jste psali o pravidle a použili symbol =>, nebo naopak psali o derivaci a použili symbol ->, mohli jste dostat až osm bodů. Dále jsem strhával 4 body, pokud jste namísto =>^+ v definici použili => (uvědomte si, že v libovolné gramatice platí pro každý neterminál A že A =>* A).*

Občas jste se pokusili namísto standardní definice použít nějakou ekvivalentní vlastnost necyklických gramatik, většinou však bez úspěchu. Párkrát se vyskytlo tvrzení "gramatika je necyklická, právě když neobsahuje jednoduchá a \eps-pravidla." V tomto tvrzení platí pouze jedna implikace: každá gramatika bez jednoduchých a \eps-pravidel je necyklická, ale například tato gramatika obsahuje oba tyto typy pravidel a přitom je necyklická:
S -> A
A -> \eps

Většinou jsem dával 5 bodů, pokud alespoň jedna implikace z Vašeho tvrzení byla pravdivá. Pokud jste namísto (ne)cykličnosti definovali vlastnost sebevložení, dostali jste maximálně 2 body. Za různé chybné definice mohly být 1 až 2 body za snahu, pokud v nich byla vidět alespoň vzdálená souvislost s necykličností. Za formální nedostatky jsem strhával 1-2 body.

b) Definujte, kdy je symbol bezkontextové gramatiky nepoužitelný. (10 bodů)

Správná odpověď: viz slajdy k 7. přednášce (slajd 10), či skripta (strana 66).

Bodování a časté chyby:

Zde bylo častou chybou, za kterou jsem však strhával jen 1 bod, následující tvrzení: symbol je nepoužitelný, právě když je nedosažitelný, nebo nenormovaný. To není pravda, například v následující gramatice je symbol 'a' epoužitelný a přitom je zřejmě normovaný i dosažitelný:
S -> Aa
A -> A
Problém je v tom, že 'a' se vyskytuje pouze v těch větných formách, které obsahují nenormovaný neterminál (kvůli nedostatku lepšího označení jsem si je pro sebe pojmenoval jako 'pseudonedosažitelné' symboly). Zmatek zde pravděpodobně vznikl tím, že při odstraňování nepoužitelných symbolů stačí odstranit vskutku pouze symboly nenormované a nedosažitelné. Odstraněním nenormovaných symbolů se nám "pseudonedosažitelné" (fuj) symboly změní na nedosažitelné a druhý algoritmus je již podchytí. (To je důvod, proč vždy napřed odstraňujeme nenormované neterminály, a až pak nedosažitelné symboly.)

Jinak jsem těch 10 bodů z příkladu rovnoměrně rozdělil mezi (pseudo)nedosažitelnost a nenormovanost (často Vám jedna z nich scházela). Za každou formální chybu, kvůli které nebylo řešení dobře, byl odečten 1 bod (pokud se však Vaše definice stejně jako v prvním příkladě týkala výlučně pravidel a nikoliv odvození, strhával jsem víc). Za různé špatné či okecávací definice byly maximálně 4 body dle toho, jak moc se blížily správné definici. Mimochodem, často se objevovala varianta tohoto zmateného tvrzení:"Neterminál je nenormovaný, pokud se vždy po nějaké době přepíše na něco, co jej obsahuje." Ať už si pod tím představíte cokoliv, nebude to pravda, viz následující gramatika, kde je neterminál S nenormovaný:
S -> A
A -> A

c) Definujte, kdy se gramatika nazývá kontextová. (10 bodů)

Správná odpověď: viz slajdy k 1. přednášce (slajd 25), či skripta (strana 6).

Bodování a časté chyby:

7 bodů bylo za správnou podmínku na tvar pravidel, 3 za výjimku týkající se \eps-pravidel. Pokud jste u \eps-pravidel zapomněli na to, že S se nesmí objevit na pravé straně žádného pravidla, strhl jsem 2 body. Pokud jste v definici namísto neostré nerovnosti použili ostrou nerovnost (to by pak v gramatice nebyla odvoditelná slova délky 1), strhl jsem rovněž 2 body. Pokud jste napsali, že pravé strany pravidel nesmí být delší než levé strany, strhl jsem 4 body (v takovém případě by naopak gramatika mohla generovat jen slova délky nejvýše 1).
Jeden bod jsem strhl za tvrzení, že levé strany pravidel jsou prvkem V (ono je v podstatě pravda, ale ten zápis vyvolával dojem, že na levé straně nemusí být neterminál, neboť nebylo jasné, jakým způsobem je to tvrzení kvantifikováno; pokud by někomu k lepší známce chyběl 1 bod, zde je patrně jistá možnost ho vyhádat). Pokud jste tvrdili, že na levé straně musí být řetězecneterminálů, strhával jsem 3 body. Pokud jste definovali jinou třídu gramatik, mohli jste dostat maximálně body za \eps-pravidla (Chomského hierarchii byste snad už zvládnout mohli). Poněkud tristní bylo, že definice některých z Vás protiřečily definici gramatiky jako takové.*

d) Jak se nazývá třída jazyků akceptovaná úplnými Turingovými stroji? (4 body)

Správná odpověď: třída rekursivních jazyků.

Bodování a časté chyby: Za odpovědi "třída rekursivně spočetných jazyků" či "jazyky typu 0" (nikoliv však "frázové jazyky") byly 2 body. Za odpovědi "kontextové jazyky" či "jazyky typu 1" byl 1 bod (taky se to týká Turingových strojů, ale lineárně ohraničených). Za ostatní odpovědi bylo 0 bodů.
Pokud jste se nenechali odradit délkou příspěvku a dočetli až sem, tak ještě poznamenám, že kombinace různých chyb byly hodnoceny mírněji než prostým součtem penalizací. Stejně jako u podobného příkladu na vnitrosemestrálce mohu konstatovat, že příklad nedopadl špatně, nicméně mohl dopadnout mnohem lépe (vždyť jde "jen" o definice).

Příklad 1 (zkouška 30.1.2006)

Zadání: Uveďte jazyk, který je bezkontextový, ale není regulární. Dokažte obě jeho vlastnosti.

Řešení je mnoho, například:
L={a^n.b^n | n>0}
je bezkontextový, neboť je to jazyk generovaný gramatikou
G=({S},{a,b},P,S), P={S -> ab | aSb}.
Jazyk rovněž nesplňuje lemma o vkládání pro regulární jazyky, není tedy regulární. Skutečně, buď n libovolné přirozené číslo. Zvolíme slovo w=a^n.b^n ležící v jazyce L. Uvažujme libovolné rozdělení w=xyz takové, že y není prázdné a |xy|<= n. Nutně x=a^k, y=a^l, z=a^(n-k-l).b^n, kde l>0, n>=k+l. Slovo xz pak jistě neleží v L, tedy i=0 je vhodná "pumpovací konstanta" dokazující, že L nesplňuje lemma o vkládání.

Mnoho z Vás s úspěchem k důkazu neregularity použilo i Myhill-Nerodovu větu.

Celkem za příklad bylo 45 bodů, z toho 5+15 za nalezení vhodného jazyka a důkaz, že je bezkontextový, dalších 25 za důkaz neregularity.
Častou chybou v důkazu porušení podmínky lemmatu o vkládání bylo vynechání některých rozdělení z diskuze. Někteří uvedli jakž takž znění negace PL, ale zapomněli dokázat, že jazyk ji splňuje (a proč). Za to ovšem moc bodů nedostali. Jiní použili lemma o vkládání pro bezkontextové jazyky a často dokonce "dokázali", že jimi uvedený bezkontextový jazyk jej nesplňuje.

Pokud někdo uvedl jazyk zapsaný jako množinu a pak dodal automat či gramatiku, která akceptovala jiný, bylo mu stženo až 5 bodů. Nejčastěji ovšem prošel bez ztráty, to když se jazyky lišily v nějaké drobnosti (například o prázdné slovo). Přitom za správnou odpověď lze považovat i jazyk uvedený jen jeho gramatikou (automatem). Pokud ovšem zadáte jazyk množinově i gramatikou (automatem), měly by být stejné.

V jednom řešení se místo důkazu neregularity výše popsaného jazyka L objevil odkaz do skript. Vzhledem k tomu, že v zadání se důkaz neregularity výslovně požadoval, nebyla tato odpověď akceptována.

Na závěr bych chtěl poděkovat všem, kteří se na písemku dobře naučili a ušetřili tak práci opravujícím, stres sobě a kdovíkomu dalšímu kdovíco dalšího. :)

23. 1. 2006
Příklad 5

Především je vhodné umět porozumět zadání. V příkladu 5 se po vás chtělo a) pro každou CF gramatiku ukázat čísla m a n taková, že b) L(G) je nekonečný PRÁVĚ, KDYŽ existuje w z L(G), že m < |w| <= n

To především znamená, že čísla m a n mohou záviset pouze na gramatice G, nikoli na slově w a že dokazované tvrzení je EKVIVALENCE, kteréžto důkaz se většinou předvádí ve formě důkazu DVOU IMPLIKACÍ. Je smutné, jak málo z vás si toto uvědomuje (většina(*) vůbec neukázala, jak zkonstruovat čísla m, n a brala je jako daná, stejně tak většina(*) dokazovala jen jednu implikaci).
(*) většina z těch, co alespoň něco napsali

Bodoval jsem takto:
a) ukázat, jak zkonstruovat čísla m a n - 10b
b) důkaz ekvivalence - za každou implikaci 15b, dohromady 30b

Samozřejmě, že část b) bez části a) ani nemá smysl, nicméně jsem se snažil, co mohl, abych našel alespoň nějakou myšlenku důkazu (kterou jsem hodnotil za každou implikaci 5 až 10 body, podle kvality myšlenky).

23. 1. 2006
Příklad 6

Příklad šestý mě zklamal hrozně, zvláště pak mě zklamali studenti mých cvičení, kterým jsem extra připomínal, že definici FI a FO po nich můžeme na písemce chtít a kterým jsem extra zdůrazňoval, že typy funkcí FI a FO deklarované v učebních materiálech jsou špatně. Nevím, jak tomu bylo ve cvičeních mých kolegů.

Aby bylo tedy jasno: (u značí sjednocení, P potenční množinu)
Typ funkce FIRST_1 je (N u \Sigma)* -> P (\Sigma u {\epsilon})
Typ funkce FOLLOW_1 je N -> P (\Sigma u {\epsilon})
(naprostá většina těch, kteří se o deklarování typu pokusili, to má špatně, většinou chybí "P", můžete si rozmyslet, proč je to špatně)

Hodnotil jsem takto:

a) typ funkce FIRST - 3b
b) definice funkce FIRST - 9b
c) typ funkce FOLLOW - 4b
d) definice funkce FOLLOW - 9b

V částech b) a d) jsem dával pouze jeden až dva body, bylo-li místo definice užito příkladu, postupu napočítávání, nebo slohové práce zvící celé A4 obsahující ne víc než náznak myšlenky, jak by daná fce mohla vypadat.
Dále jsem v části b) a d) strhával čtyři body tehdy, pokud vámi definovaná funkce nemohla vracet množinu obsahující epsilon. Jestli kdy budou nějaké konzultace, poznáte to snadno podle červeného nápisu "a co \epsilon?".
Čtyři body jsem taky strhával za definici typu
FI(\alpha) = {a | \alpha => aw, kde a je z \Sigma sjednoceno s {\epsilon}, w ze \Sigma*} apod.*
(za DÚ si rozmyslete proč je to špatně)
Pokud byla funkce First definována pouze pro jeden neterminál (nikoli pro řetězec terminálů a neterminálů), strhával jsem 3 body.
Za drobné chyby v definicích jsem strhával jeden až dva body, v závislosti na závažnosti chyby.

23. 1. 2006
Příklad

Dokažte, že pro každou CFG G se dají sestrojit čísla m, n taková, že platí: L(G) je nekonečný <=> existuje w z L(G) tak, že m < |w| <= n.

Myšlenky důkazu jsou asi tři, ta první je hlavní:
a) užijeme PL to tak, že za m zvolíme p (ze znění PL) a za n = p + q
b) dokazujeme EKVIVALENCI, to jest DVĚ IMPLIKACE a v obou se nějak použije PL
c) jedna IMPLIKACE (tahle: <=) vůbec nevyužívá toho "n"

Důkaz:
Je-li G CFG, pak pro L(G) platí Pumping Lemma. Zvole tedy m = p a n = p + q, kde p, q jsou z Pumping Lemmatu.

(<=) Nechť existuje w z L(G) takové, že m < |w| <= n. Chceme ukázat, že L(G) je nekonečný. Ale z PL víme, že pro každé slovo delší než p (= m), existuje rozdělění w = uvzxy (kde vx není epsilon), že u(v^i)z(x^i)y je taky z L pro každé i nezáporné celé. Tedy L(G) obsahuje nekonečně mnoho slov (pro všechna i jsou tato slova po dvou různá). QED

(=>) Nechť L(G) je nekonečný. Chceme najít slovo w z L(G) takové, že m < |w| <= n. Ale protože L(G) je nekonečný, obsahuje pro každé číslo k nekonečně slov délky větší než k, tedy to speciálně platí i pro m. Ze všech slov delších než m vybereme to nejkratší a označíme ho w. Nyní samozřejmě w splňuje m < |w| a chceme dokázat, že |w| <= n.

To dokážeme sporem. Budeme předpokládat, že |w| > n. Protože n > m = p, z PL víme, že existuje rozdělení w = uvzxy s vlastnostmi: vx není epsilon, |vzx| <= q takové, že pro každé i je i u(v^i)z(x^i)y slovem z L(G). Protože jsem zvolili n = p + q, pak víme, že |uzy| = |w| - |vx| > (p + q) - q = p = m a zároveň |uzy| < |w| (protože vx není epsilon). Dohromady nám to dává fakt, že uzy (= u(v^0)z(x^0)y) je délky větší než m, ale je kratší než w, což je spor s minimalitou w. QED

13. 1. 2005
Příklad

Na písemce se objevil příklad 4.)
a)Uveď příklad CFG jenž (i) není regulární, ale generuje regulární jazyk (ii). Obě tvrzení (i,ii) dokažte.
b)Př. CFG , jenž generuje jazyk jenž není CFL. Zdůvodni.

a)
G = ({S,A},{a,b},P,S)
P = {
S -> aAb
A -> a }

L(G) = {aab} - coz je regularni jazyk (lze sestrojit FA rozpoznavajici tento jazyk)

b)
Takova gramatika neexistuje! Viz Chomskeho hiearchie jazyku a gramatik.