Sveučilište u Zagrebu

Fakultet elektrotehnike i računarstva

**Ocjena:9/10**

**Druga domaća zadaća iz predmeta**

“**Uvod u teoriju računarstva**”

Zadatak broj 2042

Zagreb, lipanj 2010.

**Druga domaća zadaća iz predmeta “Uvod u teoriju računarstva”**

**Student:**

**Matični broj studenta:**

**Zadatak broj 2042:**

Jezik L nad abecedom {a,b,c,d} čine prazan niz i svi mogući nizovi u kojima je broj znakova a, b, c i d jednak, odnosno na = nb = nc =nd . DKA M(m) prihvaća sve nizove iz jezika L koji su oblika w1w2w3.....wk  , pri čemu su wi ϵ L i | wi |<=4\*m. Konstruirati minimalne DKA M(1) i DKA M(2). Odrediti broj stanja minimalnog DKA M(m) u ovisnosti o paramatru m.

**Uvod**

Problematika zadatka je relativno jednostavna. Potrebno je konstruirati deterministički konačni automat za parametre m=1 i m=2 koji će ispunjavati zadane uvjete:

1. Niz je element jezika L ako je satavljen od znakova a,b,c,d te ako je broj tih znakova jednak (na = nb = nc =nd)
2. DKA(m) prihvaća niz ako je niz podijeljen na blokove koji su elementi jezika L i čija je duljina bloka manja ili jednaka parametru m pomnoženom s 4 (|wi|<=4\*m)

Problem ću rješiti crtanjem automata na papiru, a zatim napraviti slike na računalu uz pomoć napravljenih skica.

Deterministički konačni automat čini konačni skup stanja i funkcije prijelaza. Funkcija prijelaza jednoznačno je određena stanjem u kojem je automat i znakom na ulazu. Konačni automat počinje rad u početnom stanju, a prijeđe li za neki ulazni niz x u jedno od prihvatljivih stanja tada se taj niz x prihvaća.

Formalno se zadaje uređenom petorkom dka=(Q,∑,δ,q0,F) gdje je:

Q – konačan skup stanja

∑ - konačan skup ulaznih znakova

δ – funkcija prijelaza Qx∑ → Q

q0 ϵ Q – početno stanje

F – skup prihvatljivih stanja

**Ostvarenje**

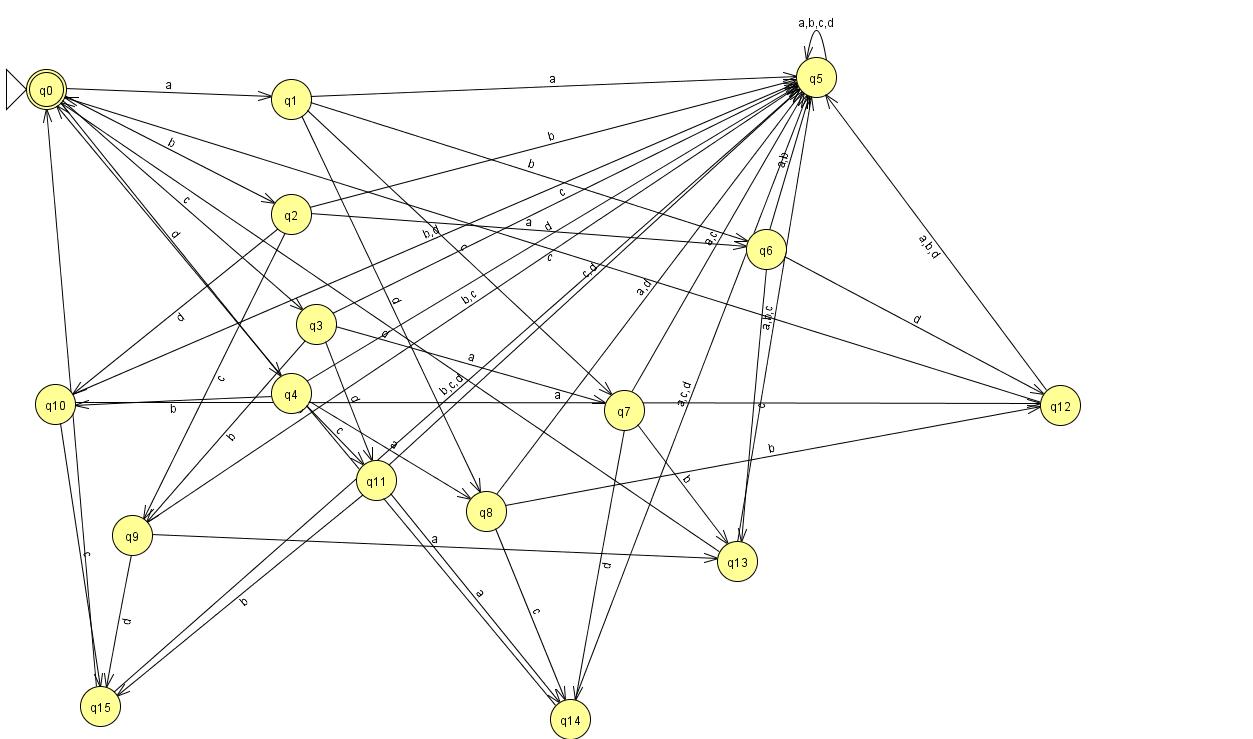
**Konstrukcija automata DKA M(1)**

Za automat DKA M(1) vrijedi da je parametar m = 1. Duljina bloka koji mora biti element jezika L je tada |wi|<=4. S obzirom na to da jezik L sadrži i prazan niz, duljina bloka može biti 0. Uvjet na = nb = nc =nd onemogućava prihvaćanje blokova od jednog, dva ili tri znaka, te je zadovoljen jedino za blok od četiri znaka.

Dakle, prihvaćaju se nizovi znakova a,b,c,d podijeljeni u blokove po 4 znaka, s tim da se u svakom bloku znak a pojavi jednom, znak b jednom, znak c jednom i znak d jednom.

Prihvaćanje praznog niza osigurao sam postavljanjem prihvatljivog početnog stanja. Zatim je uslijedilo ispitivanje prihvatljivosti za sve nizove sastavljene od 4 znaka. Ukoliko je neki slijed nakon pročitanog četvrtog znaka prihvatljiv tada se gradi prijelaz iz trenutnog stanja u početno prihvatljivo stanje za odgovarajući znak, te postupak kreće iznova za novi blok od četiri znaka.

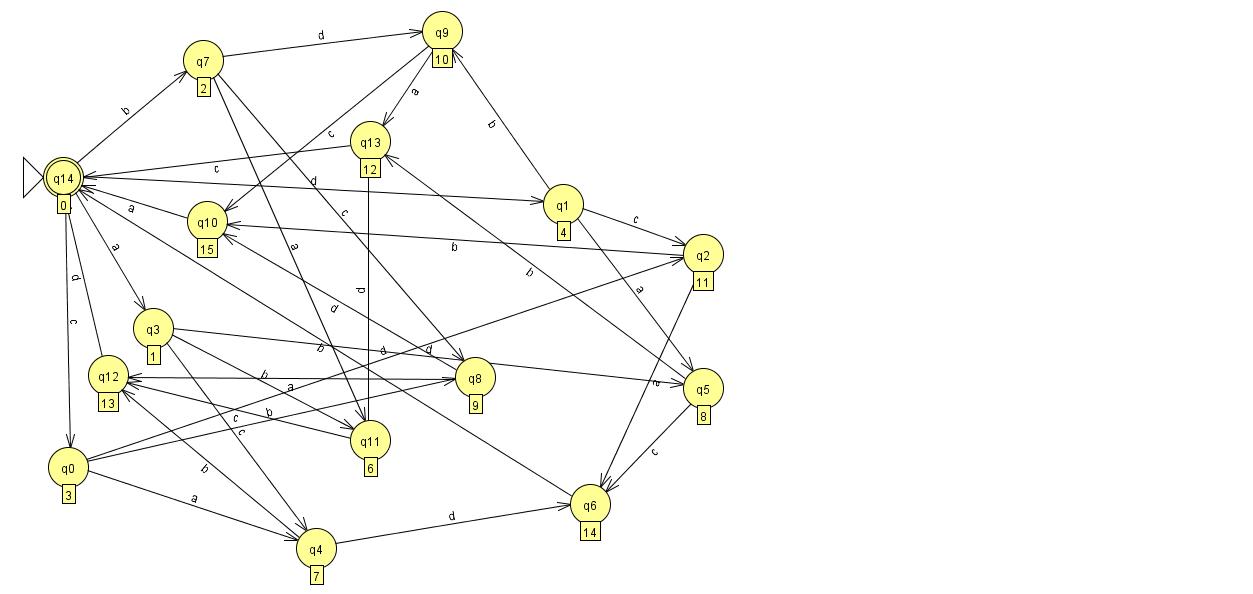
Slika 1:



Na slici 1 je prikazan DKA M(1). Prilikom konstukcije sam ga prvo nacrtao na papiru, a zatim kao pomoć koristio program JFLAP u kojem sam napravio ovu sliku. Početno stanje je označeno strelicom. Prihvatljivo stanje je zaokruženo dva puta.

\*veća slika je u dodatku (vrijedi i za ostale slike u ovom dokumentu)

Slika 2:



Na slici 2 nalazi se minimizirani automat DKA M(1). Jedino što ga razlikuje od neminimiziranog sa slike 1 je izbacivanje stanja q5 koje je meni sluzilo za prikupljanje svih neprihvatljivih prijelaza. To stanje sam koristio zbog lakšeg snalaženja prilikom izrade na papiru, ali je nepotrebno jer su svakako sva stanja osim početnog neprihvatljiva. U kvadratima pridruženim stanjima sa slike 2 se nalaze brojevi koji pokazuju koje stanje sa slike 1 dotično stanje predstavlja.

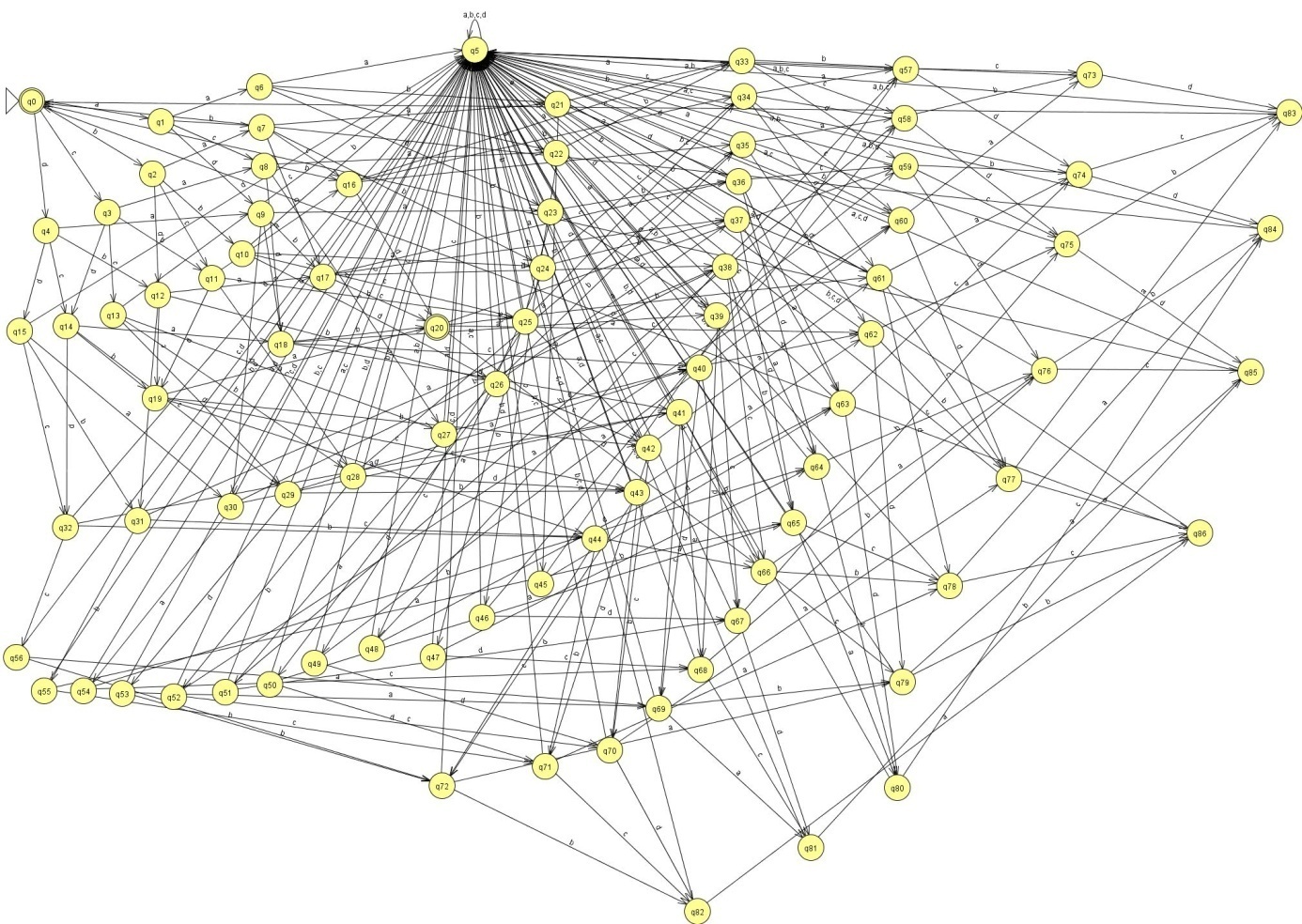
**Konstrukcija automata DKA M(2)**

Za parametar m=2 konstruira se DKA M(2) koji je daleko složeniji nego DKA M(1) iz proslog dijela zadatka. Sada se niz dijeli na blokove velicine 4 ili 8 znakova zbog uvjeta |wi|<=8. Dio automata koji će ispitivati blokove od 4 znaka je dosta sličan DKA M(1) s uvedenim novim prihvatljivim stanjem (koje omogućava da prihvatljivi blok od 4 znaka bude i sufiks nekom bloku od 8 znakova). Izrada je bila komplicirana zbog mogućnosti sitne pogreške koja bi rezultirala potpuno pogrešnim rješenjem. Princip izrade je isti kao kod DKA M(1), samo što DKA M(2) mora ispitivati i nizove od 8 znakova što je rezultiralo daleko većim brojem stanja. U blokovima od 8 znakova koji se prihvaćaju mora vrijediti na = nb = nc =nd = 2.

Početno stanje je označeno strelicom na slici, a dva su stanja prihvatljiva. Početno prihvaća prazan niz i blokove od 8 znakova koji ispunjavaju uvjet dok stanje q20 mora također biti prihvatljivo kako bi se prihvatili blokovi od 4 znaka u kojima vrijedi na = nb = nc =nd = 1. Za nizove od 4 znaka automat provjeri dva bloka i vraća se u početno stanje ako su prihvatljivi te ide na sljedeće blokove. Za nizove od 8 znakova provjeri jedan blok te se vrati u početno stanje ako je prihvatljiv i uzima drugi blok na provjeru.

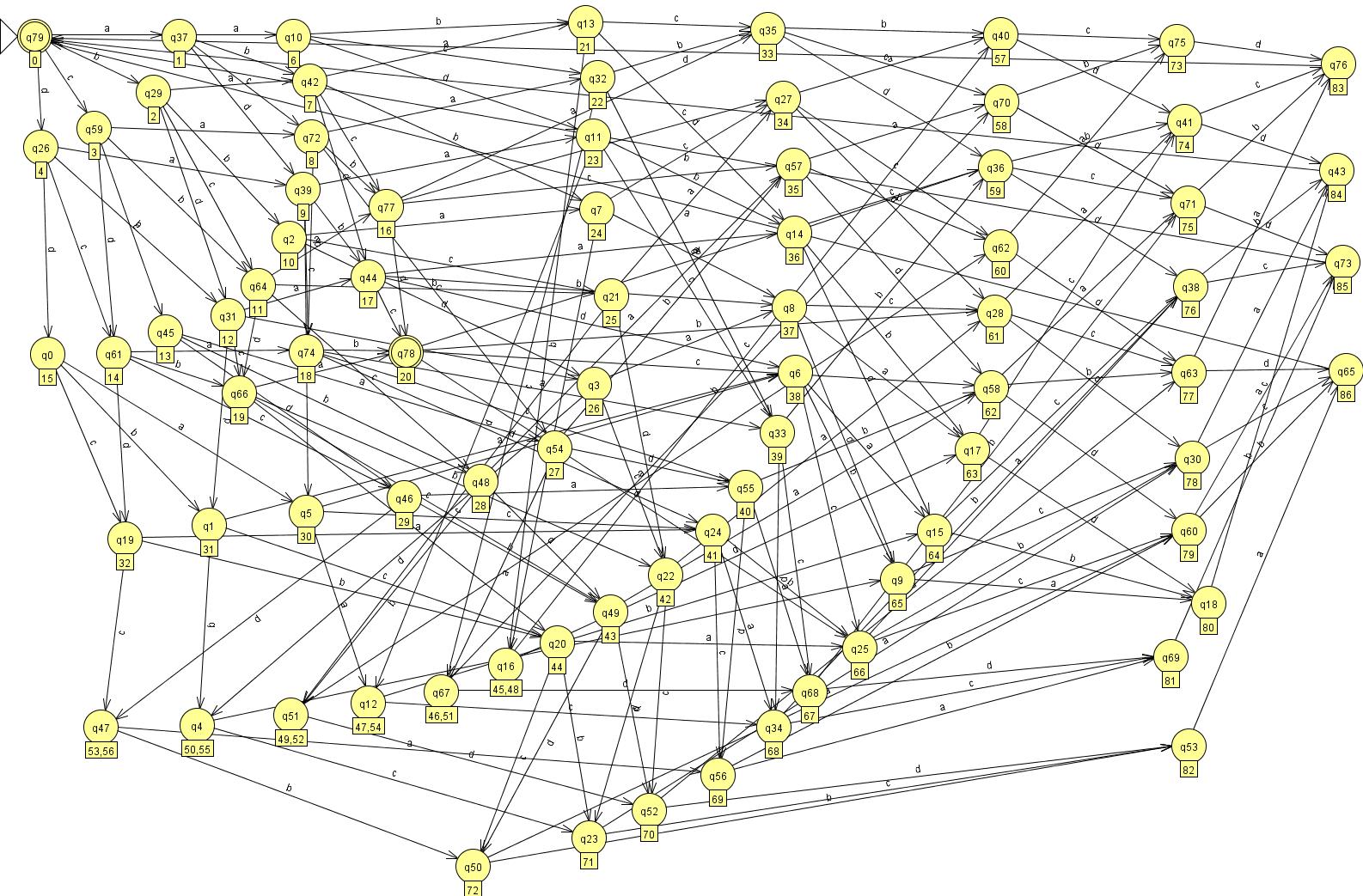
Na slici 3 je moj neminimizirani DKA M(2) koji ima 87 stanja od kojih su dva prihvatljiva.

Slika 3:



Na slici 4 je minimizirani DKA M(2) koji ima 80 stanja. Stanja koja su izbačena iz mog prvobitnog modela su q5 (koje mi je i ovdje kao i kod DKA(1) koristilo samo za prikupljanje svih neprihvatljivih prijelaza), zatim stanja q48, q51, q52, q54, g55 i g56 koja su redom istovjetna sa stanjima q45, q46, q49, q47, q50, q53. Ako uzmem za primjer par stanja q48 i q45 ona su istovjetna jer je u neminimiziranom DKA(2) stanje q48 prihvaćalo niz bbaa a stanje q45 niz aabb što su dva jednaka niza s obzirom na to da poredak znakova nema nikakvog utjecaja na uvjete zadane u zadatku, važan je samo broj njihovih pojavljivanja. Istu grešku san napravio i kod ostalih pet parova istovjetnih stanja. Minimizacija je napravljena pomoću JFLAP-a.

Slika 4:



**Određivanje broja stanja minimalnog DKA M(m) u ovisnosti o parametru m**

Broj stanja minimalnog DKA M(m) određen je funkcijom:

Za m=1 broj stanja je 15

Za m=2 broj stanja je 80

Za m=3 broj stanja je 255

Za m=4 broj stanja je 624

.....

Funkciju sam dobio pomoću opcije interpolate u Wolfram alpha u koji sam ubacio točke i dobio funkciju. Točke su izračunate java algoritmom koji zbraja sve kombinacije znakova a,b,c,d za duljinu niza od 1 do 4\*m te izbacuje sve one u kojima se neki znak previše puta pojavio, te one koje su jednake (jednake su ako imaju jednak broj svih znakova npr AABC i BACA). Postupak se ponavlja po razinama.

**Zaključak**

Prilikom izrade ove domaće zadaće najveći problem predstavljala je izrada DKA(2) zbog jako velikog broja stanja. Nakon nekog vremena crtanje postaje nepregledno zbog velikog broja prijelaza koji se isprepleću te otežavaju praćenje osnovne ideje rješavanja. U slučaju da je trebalo napraviti i automat za parametar m=3 posao bi se previse zakomplicirao jer bi broj stanja porastao na 255 što bi bilo prezahtjevno za praćenje na crtežu. Ne bi bilo neizvedivo ali bi zahtjevalo dosta vremena i koncentracije i veliku površinu za crtanje koja bi barem donekle osigurala preglednost i omogućila konačan dolazak do točnog rješenja.