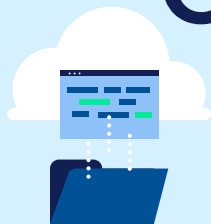


METODE KLASA EKVIVALENCIJE I GRANIČNIH VREDNOSTI

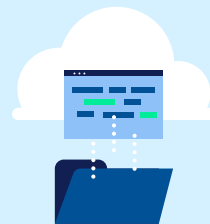


TESTIRANJE SOFTVERA

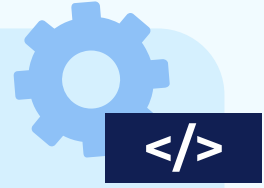


01

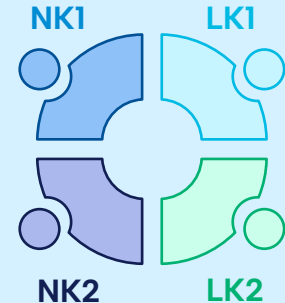
METODA KLASA EKVIVALENCIJE



KLASE EKVIVALENCIJE



- Nije praktično testirati program za svaki mogući skup ulaznih podataka.
- Ideja je da podelimo ulazne podatke na podskupove – klase ekvivalencije.
- Svi ulazni podaci koji pripadaju jednoj klasi ekvivalencije treba da daju iste ili slične rezultate pri izvršavanju programa.
- Prilikom projektovanja testova, bismo barem jednog reprezentativnog predstavnika svake klase.
- U idealnom slučaju, klase su međusobno disjunktne i pokrivaju čitav skup ulaznih podataka.
- Za svaki uslov posmatraju se dve grupe klasa ekvivalencije:
 - **Legalne klase** – obuhvataju dozvoljene situacije
 - **Nelegalne klase** – obuhvataju sve ostale situacije

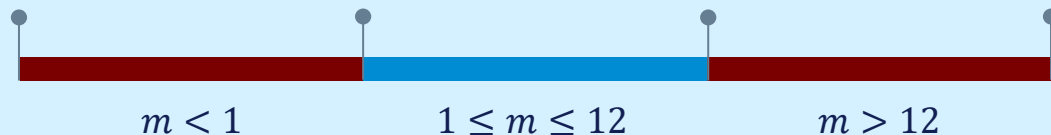


ODREĐIVANJE KLASA EKVIVALENCIJE



USLOV ZA OPSEG

- U opštem slučaju, ako je za ulazni podatak p dozvoljen opseg vrednosti $[a, b]$, definišu se:
 - Jedna legalna klasa za sam opseg ($a \leq p \leq b$)
 - Jedna nelegalna klasa ispod opsega ($p < a$)
 - Jedna nelegalna klasa iznad opsega ($p > b$)
- Primer – Unos meseca kao celog broja od 1 do 12 u formi za registraciju.
 - Klase ekvivalencije:



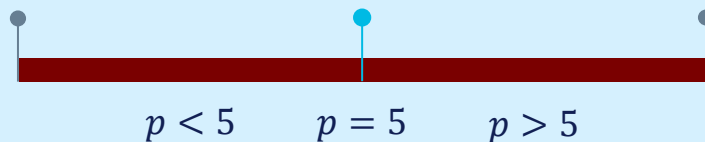
- Ispravni podaci: 3, 7
- Neispravni podaci: -2, 0, 15

ODREĐIVANJE KLASA EKVIVALENCIJE

USLOV ZA DUŽINU



- Ako je za ulazni podatak p određeno da mora biti dužine d , definišu se:
 - Jedna legalna klasa za tačnu dužinu ($p = d$)
 - Jedna nelegalna klasa za dužinu manju od definisane ($p < d$)
 - Jedna nelegalna klasa za dužinu veću od definisane ($p > d$)
- Primer – Unos poštanskog broja kao petocifrenog broja u formi veb prodavnice.
 - Klase ekvivalencije:



- Ispravni podaci: 11120, 58274
- Neispravni podaci: 123, 0, 458291, prazno polje

ODREĐIVANJE KLASA EKVIVALENCIJE



USLOV ZA NABRAJANJE

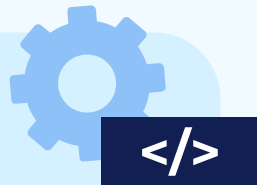
- Ako je za ulazni podatak p dozvoljen skup vrednosti $v = \{v_1, v_2, \dots, v_n\}$, definišu se:
 - Po jedna legalna klasa za svaku vrednost iz ovog skupa **za koju se program različito ponaša** ($p = v_1, \dots, p = v_n$)
 - Jedna nelegalna klasa za sve vrednosti van skupa ($p \notin v$)
- Primer – Unos krvne grupe u formi veb aplikacije zdravstvene ustanove.
 - Klase ekvivalencije:



- Ispravni podaci: A, B, O
- Neispravni podaci: 0, C, nulta

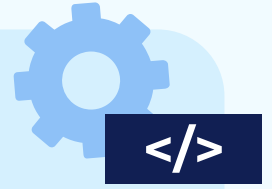
ODREĐIVANJE KLASA EKVIVALENCIJE

USLOV ZA OBAVEZNOST

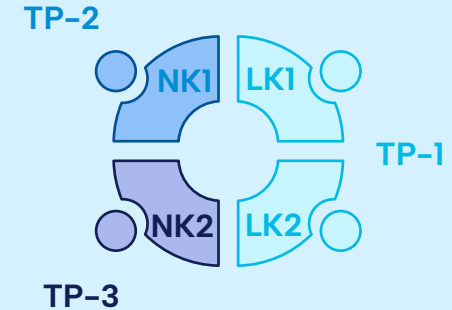


- Ako je za ulazni podatak p obavezno da mora zadovoljiti format f , definišu se:
 - Najmanje jedna legalna klasa za ispravan format
 - Najmanje jedna nelegalna klasa za neispravan format
- Primer – Unos mejl adrese pri registraciji u formi veb prodavnice.
 - Klase ekvivalencije:
 - Mejl adresa mora da ima prefiks, simbol „@“, server, tačku i domen.
 - Mejl adresa nema simbol „@“.
 - Mejl adresa nema tačku.
 - Mejl adresa nema prefiks.
 - Mejl adresa nema domen.
 - Ispravni podaci: ts@etf.rs, nik0la@gmail.com
 - Neispravni podaci: abc.rs, ts@etfrs, @etf.rs, ts@etf

PROJEKTOVANJE TESTOVA



- Prvo je neophodno definisati i numerisati sve legalne i nelegalne klase ekvivalencije.
- Potom, potrebno je projektovati test primere. Svaki test primer ima ulazne podatke, očekivani izlaz i klase ekvivalencije koje taj test primer pokriva.
- Sve klase ekvivalencije treba pokriti test primerima:
 - Što više legalnih klasa treba pokriti sa što manje test primera.
 - Za svaku nelegalnu klasu treba projektovati po jedan test primer.
 - Zašto ne pokriti više nelegalnih klasa u jednom test primeru?



ZADATAK 1 – PRODAVNICA OBUĆE

POSTAVKA



- Neka je dat deo softverskog sistema prodavnice obuće koji služi za unos artikala i spisak različitih veličina određene obuće. Specifikacija tekstualne datoteke koja se učitava u program navodi da artikli treba da budu alfabetski znaci od 2 do 15 karaktera. Svaka veličina može biti u opsegu od 26 do 55, i predstavlja se samo celim brojevima. Veličine se unose u rastućem poretku (prvo manje veličine). Maksimalno pet veličina se može uneti za svaki artikal. Format unosa je: naziv artikla, sledi zarez, a zatim sledi spisak veličina. Zarez se takođe koristi da razdvoji veličine. Blanko znaci treba da se zanemare na ulazu.
1. Napisati sve klase ekvivalencije za ovaj deo softverskog sistema.
 2. Napisati skup test primera koji će pokriti definisane klase ekvivalencije (za svaki test primer navesti ulazne podatke, očekivani izlaz i klase ekvivalencije koje on pokriva).



ZADATAK 1 – PRODAVNICA OBUĆE

REŠENJE – KLASA EKVIVALENCIJE (1)



● Naziv artikla:

- Naziv artikla sadrži samo slova
- Naziv artikla sadrži karaktere koji nisu slova
- Dužina naziva artikla je manja od 2
- Dužina naziva artikla je između 2 i 15
- Dužina naziva artikla je veća od 15

● Veličina:

- Veličina je manja od 26
- Veličina je u opsegu od 26 do 55
- Veličina je veća od 55
- Veličina sadrži samo cifre
- Veličina sadrži karaktere koji nisu cifre
- Veličina je ceo broj
- Veličina je realan broj

● Lista veličina:

- Veličine su unete u rastućem redosledu
- Veličine su unete u opadajućem redosledu
- Nisu unete veličine za neki artikal
- Uneto je između jedne i pet veličina za artikal
- Uneto je više od pet veličina za artikal

● Format unosa:

- Naziv artikla je unet pre veličine
- Naziv artikla nije unet pre veličine
- Svaki podatak je razdvojen zarezom
- Svaki podatak nije razdvojen zarezom
- Ulazni podaci ne sadrže blanko znakove (???)
- Ulazni podaci sadrže blanko znakove (???)

ZADATAK 1 – PRODAVNICA OBUĆE

REŠENJE – KLASA EKVIVALENCIJE (2)



● Naziv artikla:

1. (LK) Naziv artikla sadrži samo slova
2. (NK) Naziv artikla sadrži karaktere koji nisu slova
3. (NK) Dužina naziva artikla je manja od 2
4. (LK) Dužina naziva artikla je između 2 i 15
5. (NK) Dužina naziva artikla je veća od 15

● Veličina:

6. (NK) Veličina je manja od 26
7. (LK) Veličina je u opsegu od 26 do 55
8. (NK) Veličina je veća od 55
9. (LK) Veličina sadrži samo cifre
10. (NK) Veličina sadrži karaktere koji nisu cifre
11. (LK) Veličina je ceo broj
12. (NK) Veličina je realan broj

● Lista veličina:

13. (LK) Veličine su unete u rastućem redosledu
14. (NK) Veličine su unete u opadajućem redosledu
15. (NK) Nisu unete veličine za neki artikal
16. (LK) Uneto je između jedne i pet veličina za artikal
17. (NK) Uneto je više od pet veličina za artikal

● Format unosa:

18. (LK) Naziv artikla je unet pre veličine
19. (NK) Naziv artikla nije unet pre veličine
20. (LK) Svaki podatak je razdvojen zarezom
21. (NK) Svaki podatak nije razdvojen zarezom
22. (LK) Ulazni podaci ne sadrže blanko znakove
23. (LK) Ulazni podaci sadrže blanko znakove

ZADATAK 1 – PRODAVNICA OBUĆE

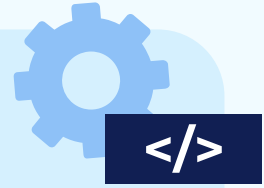
REŠENJE – TEST PRIMERI



TP	Ulazna datoteka	Očekivani izlaz	Pokrivene klase ekvivalencije
1	reebok,40,43	Ispravno	1, 4, 7, 9, 11, 13, 16, 18, 20, 22
2	AbcDeFghijKlmno,41,42,43, 44, 45	Ispravno	1, 4, 7, 9, 11, 13, 16, 18, 20, 23
3	8merrell,36,38	Neispravno	2
4	M,36,38	Neispravno	3
5	AbcDeFghijKlmnoprsT,36,38	Neispravno	5
6	Xy,6	Neispravno	6
7	Xy,66	Neispravno	8
8	ABC,2?7	Neispravno	10
9	XY,2.5	Neispravno	12
10	abc,32,31,33,34,28	Neispravno	14
11	nike,	Neispravno	15
12	nike,41,42,43,44,45,46	Neispravno	17
13	31,32,33,34,35	Neispravno	19
14	XY32,33,34,35	Neispravno	21

ZADATAK 2 – TROUGLOVI

POSTAVKA



- Neka je dat program koji učitava vrednosti iz 3 tekstualna polja (poljeA, poljeB i poljeC). U svako tekstualno polje korisnik treba da unese celobrojnu vrednost od 1 do 50, koja predstavlja dužinu stranice trougla. Unos alfabetskih karaktera ili brojeva drugog tipa nije zabranjen. Kada se klikne na dugme za potvrdu, funkcija **proveri_trougao** ovog programa vraća kategoriju kojoj trougao pripada, odnosno jedan od sledećih izlaznih podataka: „Nije trougao“, „Raznostrani“, „Jednakokraki“ ili „Jednakostranični“. Takođe, ukoliko je trougao moguće formirati, na izlazu se pojavljuje vrednost obima tog trougla. U suprotnom, prikazuje se poruka o grešci.
- 1. Napisati sve klase ekvivalencije za ulazne i izlazne podatke u skladu sa specifikacijom.
- 2. Napisati skup test primera **korišćenjem minimalnog pristupa** koji će pokriti definisane klase ekvivalencije (za svaki test primer navesti ulazne podatke, očekivani izlaz i klase ekvivalencije koje on pokriva).





ZADATAK 2 – TROUGLOVI

REŠENJE – KLASA EKVIVALENCIJE ZA ULAZNE PODATKE

- **Polje „poljeA“:**

1. (LK) U poljeA je unet ceo broj između 1 i 50
2. (NK) U poljeA je unet ceo broj manji od 1
3. (NK) U poljeA je unet ceo broj veći od 50
4. (NK) U poljeA je unet realan broj
5. (NK) U poljeA je unet alfabetski karakter

- **Polje „poljeB“:**

6. (LK) U poljeB je unet ceo broj između 1 i 50
7. (NK) U poljeB je unet ceo broj manji od 1
8. (NK) U poljeB je unet ceo broj veći od 50
9. (NK) U poljeB je unet realan broj
10. (NK) U poljeB je unet alfabetski karakter

- **Polje „poljeC“:**

11. (LK) U poljeC je unet ceo broj između 1 i 50
12. (NK) U poljeC je unet ceo broj manji od 1
13. (NK) U poljeC je unet ceo broj veći od 50
14. (NK) U poljeC je unet realan broj
15. (NK) U poljeC je unet alfabetski karakter



ZADATAK 2 – TROUGLOVI

REŠENJE – KLASA EKVIVALENCIJE ZA IZLAZNE PODATKE

- **Izlaz programa:**

1. (LK) Trougao je raznostrani i izračunat je njegov obim.
2. (LK) Trougao je jednakokraki i izračunat je njegov obim.
3. (LK) Trougao je jednakostranični i izračunat je njegov obim.
4. (LK) Nije trougao. (!)
5. (LK) Greška! Stranica A nije u dozvoljenom opsegu vrednosti.
6. (LK) Greška! Stranica B nije u dozvoljenom opsegu vrednosti.
7. (LK) Greška! Stranica C nije u dozvoljenom opsegu vrednosti.
8. (LK) Greška! Nedozvoljen ulazni podatak.

- (!) Šta ako je trougao nemoguće konstruisati?

- U specifikaciji nisu data sva pravila konstruisanja trougla, ali se proverava da li je trougao uopšte moguće konstruisati prirodno nameće.
- Pravilo kaže da zbir dužina svake dve stranice mora biti manji od dužine treće stranice.
- Ne možemo da znamo da li je ova provera implementirana jer nemamo pristup programskom kodu.



ZADATAK 2 – TROUGLOVI

REŠENJE – TEST PRIMERI (1)

TP	poljeA	poljeB	poljeC	Očekivani izlaz	Pokrivene klase ekvivalencije
1	3	4	5	Raznostrani (obim je 12).	U-1, U-6, U-11, I-1
2	2	2	3	Jednakokraki (obim je 7).	U-1, U-6, U-11, I-2
3	5	5	5	Jednakostranični (obim je 15).	U-1, U-6, U-11, I-3
4	4	1	2	Nije trougao.	U-1, U-6, U-11, I-4
5	-1	5	5	Greška. Stranica A nije u opsegu.	U-2, U-6, U-11, I-5
6	51	5	5	Greška. Stranica A nije u opsegu.	U-3, U-6, U-11, I-5
7	3.5	5	5	Greška. Nedozvoljen ulaz.	U-4, U-6, U-11, I-8
8	D	5	5	Greška. Nedozvoljen ulaz.	U-5, U-6, U-11, I-8

ZADATAK 2 – TROUGLOVI

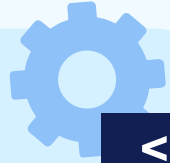
REŠENJE – TEST PRIMERI (2)



TP	poljeA	poljeB	poljeC	Očekivani izlaz	Pokrivene klase ekvivalencije
9	5	0	5	Greška. Stranica B nije u opsegu.	U-1, U-7 , U-11, I-6
10	5	74	5	Greška. Stranica B nije u opsegu.	U-1, U-8 , U-11, I-6
11	5	12.24	5	Greška. Nedoovoljen ulaz.	U-1, U-9 , U-11, I-8
12	5	@	5	Greška. Nedoovoljen ulaz.	U-1, U-10 , U-11, I-8
13	5	5	-8	Greška. Stranica C nije u opsegu.	U-1, U-6, U-12 , I-7
14	5	5	129	Greška. Stranica C nije u opsegu.	U-1, U-6, U-13 , I-7
15	5	5	6.1	Greška. Nedoovoljen ulaz.	U-1, U-6, U-14 , I-8
16	5	5	x	Greška. Nedoovoljen ulaz.	U-1, U-6, U-15 , I-8

ZADATAK 3 – OCENE

POSTAVKA (1)

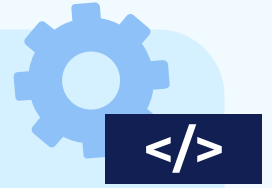


- Marko je napravio aplikaciju koja mu pomaže da odredi ocenu koju će dobiti na ispitu. Aplikacija prihvata broj poena osvojen na ispitu (celobrojna vrednost do 75) i broj poena osvojen na domaćem zadatku (celobrojna vrednost do 25), na osnovu kojih generiše ocenu na predmetu. Ocena može biti: „A“, „B“, „C“ ili „D“. Ocena se računa na osnovu ukupnog broja poena koji se dobija kao zbir poena na ispitu i domaćem zadatku:
 - Ocena „A“ – broj poena veći ili jednak 70
 - Ocena „B“ – broj poena veći ili jednak 50, ali manji od 70
 - Ocena „C“ – broj poena veći ili jednak 30, ali manji od 50
 - Ocena „D“ – broj poena manji od 30
- Ukoliko je ukupan broj poena izvan očekivanog opsega, aplikacija prikazuje poruku „Greška“.



ZADATAK 3 – OCENE

POSTAVKA (2)



1. Napisati sve klase ekvivalencije za ulazne i izlazne podatke u skladu sa specifikacijom aplikacije. Razmatrati i validne i nevalidne ulaze i izlaze.
2. Napisati skup test primera **korišćenjem pristupa „1-1“** koji će pokriti definisane klase ekvivalencije (za svaki test primer navesti ulazne podatke, očekivani izlaz i klase ekvivalencije koje on pokriva).

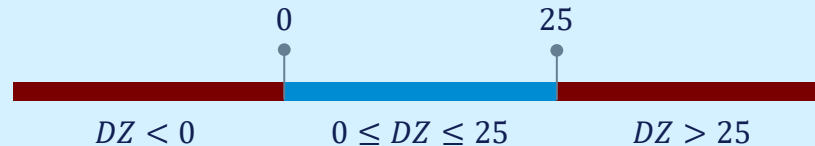
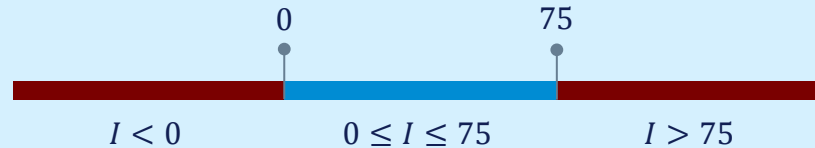


ZADATAK 3 – OCENE

REŠENJE – KLASJE EKVIVALENCIJE ZA ULAZNE PODATKE (I)



- Broj poena na ispitu i domaćem zadatku moraju biti nenegativni!
- Dakle, broj poena na ispitu (I) je celobrojna vrednost između 0 i 75, a broj poena na domaćem zadatku (DZ) celobrojna vrednost između 0 i 25.





ZADATAK 3 – OCENE

REŠENJE – KLASA EKVIVALENCIJE ZA ULAZNE PODATKE (2)

- **Ispit:**

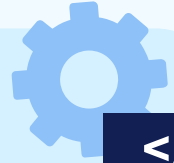
1. (LK) Broj poena na ispitu je ceo broj između 0 i 75
2. (NK) Broj poena na ispitu je ceo broj manji od 0
3. (NK) Broj poena na ispitu je ceo broj veći od 75
4. (NK) Broj poena na ispitu je realan broj
5. (NK) Broj poena na ispitu sadrži alfabetske karaktere

- **Domaći zadatak:**

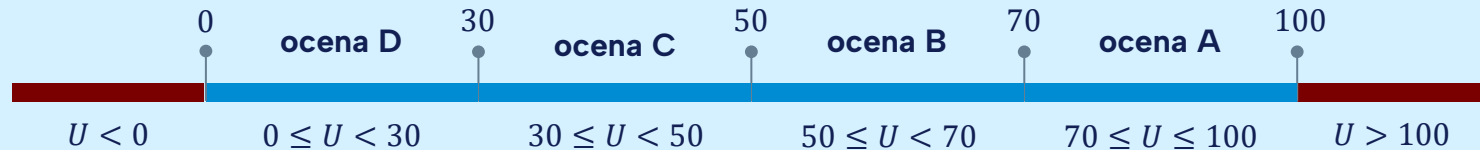
6. (LK) Broj poena na domaćem zadatku je ceo broj između 0 i 25
7. (NK) Broj poena na domaćem zadatku je ceo broj manji od 0
8. (NK) Broj poena na domaćem zadatku je ceo broj veći od 25
9. (NK) Broj poena na domaćem zadatku je realan broj
10. (NK) Broj poena na domaćem zadatku sadrži alfabetske karaktere

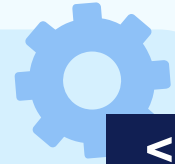
ZADATAK 3 – OCENE

REŠENJE – KLASJE EKVIVALENCIJE ZA IZLAZNE PODATKE (I)



- Ukupan broj poena (U) je nenegativna celobrojna vrednost manja od 100.
- Da li su granice opsega vrednosti samo 0 i 100?
 - Ne, granice su sve vrednosti na dozvoljenom intervalu oko kojih se menja izlaz programa.





ZADATAK 3 – OCENE

REŠENJE – KLASA EKVIVALENCIJE ZA IZLAZNE PODATKE (2)

- **Validni izlazi:**

1. (LK) Ocena je „A“.
2. (LK) Ocena je „B“.
3. (LK) Ocena je „C“.
4. (LK) Ocena je „D“.
5. (LK) Greška. (Ovo je prema specifikaciji generička greška koja obuhvata sve neispravne ulazne podatke.)

- Šta bi bili nevalidni izlazi?

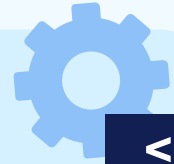
- Ovde je izbor klasa ekvivalencije subjektivan, pa jedan mogući skup nevalidnih izlaznih klasa može biti kao u nastavku.

- **Nevalidni izlazi:**

6. (NK) Ocena je „E“. (Ako je broj poena manji od 0.)
7. (NK) Ocena je „A+“. (Ako je broj poena veći od 100.)
8. (NK) Ocena je *null*. (Nedefinisan izlaz.)

ZADATAK 3 – OCENE

REŠENJE – TEST PRIMERI (1)



TP	I	DZ	U	Testirana particija	Očekivani izlaz	Pokrivene klase ekvivalencije
1	44	15	59	$0 \leq I \leq 75$	Ocena „B“	U-1 , U-6, I-2
2	-10	15	5	$I < 0$	Greška	U-2 , U-6, I-5
3	93	15	108	$I > 75$	Greška	U-3 , U-6, I-5
4	48.7	15	?	I je realan broj	Greška	U-4 , U-6, I-5
5	5x	15	?	I sadrži alf. karaktere	Greška	U-5 , U-6, I-5
6	40	8	48	$0 \leq DZ \leq 25$	Ocena „C“	U-1, U-6, I-3
7	40	-15	25	$DZ < 0$	Greška	U-1, U-7 , I-5
8	40	47	87	$DZ > 25$	Greška	U-1, U-8 , I-5
9	40	12.76	?	DZ je realan broj	Greška	U-1, U-9 , I-5
10	40	a	?	DZ sadrži alf. karaktere	Greška	U-1, U-10 , I-5

ZADATAK 3 – OCENE

REŠENJE – TEST PRIMERI (2)

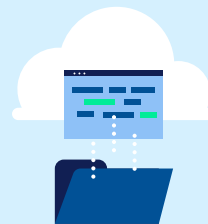


TP	I	DZ	U	Testirana particija	Očekivani izlaz	Pokrivene klase ekvivalencije
11	-10	-10	-20	$U < 0$	Greška	U-2, U-7, I-5
12	12	5	17	$0 \leq U < 30$	Ocena „D“	U-1, U-6, I-4
13	32	13	45	$30 \leq U < 50$	Ocena „C“	U-1, U-6, I-3
14	44	15	59	$50 \leq U < 70$	Ocena „B“	U-1, U-6, I-2
15	60	20	80	$70 \leq U \leq 100$	Ocena „A“	U-1, U-6, I-1
16	80	30	110	$U > 100$	Greška	U-3, U-8, I-5
17	-10	0	-10	Ocena „E“	Greška	U-2, U-6, I-6
18	100	10	110	Ocena „A+“	Greška	U-3, U-6, I-7
19	null	null	?	Ocena null	Greška	I-8



02

METODA GRANIČNIH VREDNOSTI



GRANIČNE VREDNOSTI



- Često se pri implementaciji naprave manje greške, poput zamene operatora $<$ sa $\leq$ ili $>$ sa $\geq$ i slično. Na taj način se slučajno izostave granične vrednosti.
- Primer – Specifikacija aplikacije bioskopa definiše da deca do 10 godina imaju popust na karte za bioskop.
 - Da li ovo znači da popust važi dokle god dete ne napuni 10 godina ili važi i dok ima 10 godina, odnosno do navršene 11. godine?
 - Programer nije bio siguran tokom implementacije, pa se provukla greška!
- Zbog toga je bitno obratiti pažnju na granične vrednosti.
- Metoda graničnih vrednosti predstavlja proširenje metode klasa ekvivalencije.

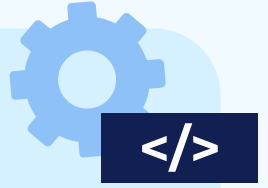


ODREĐIVANJE GRANIČNIH VREDNOSTI

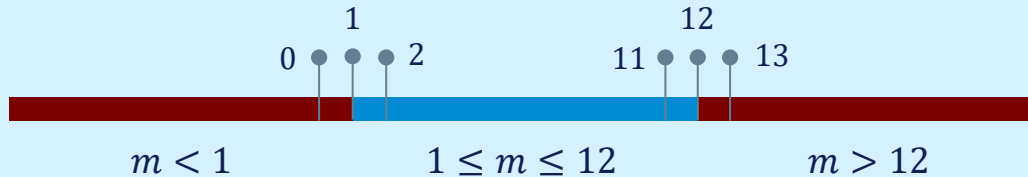


- Za svaku definisanu klasu ekvivalencije se identifikuju njene granice.
- Test primeri se projektuju tako da se za svaku granicu v odabere:
 - Jedna tačka p na samoj granici ($p = v$)
 - Jedna tačka p ispod granice ($p = v - \varepsilon$)
 - Jedna tačka p iznad granice ($p = v + \varepsilon$)
- Pomeranje u odnosu na granicu se vrši za najmanji inkrement ε :
 - Za cele brojeve je taj inkrement 1.
 - Za realne brojeve je taj inkrement najmanja razlika koja pravi promenu u programu (npr. 0.0001).
- Tačke se mogu ponoviti u okviru više klasa ekvivalencije. Zbog toga, treba pažljivo birati tačke, kako ne bi došlo do ponavljanja testova.

PROJEKTOVANJE TESTOVA



- Primer – Unos meseca kao celog broja od 1 do 12 u formi za registraciju.
 - Pored identifikacije klasa ekvivalencije, identifikuju se granice između njih (u ovom primeru to su $m = 1$ i $m = 12$).
 - Za svaku granicu projektuju se test primeri tako da oni pokriju tačku na samoj granici, ispod granice i iznad granice:
 - Za granicu $m = 1$, to su tačke 0, 1 i 2.
 - Za granicu $m = 12$, to su tačke 11, 12 i 13.



ZADATAK 4 – OCENE 2.0

POSTAVKA (1)

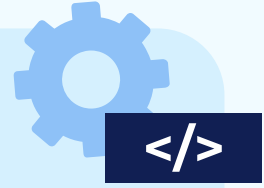


- Marko je napravio aplikaciju koja mu pomaže da odredi ocenu koju će dobiti na ispitu. Aplikacija prihvata broj poena osvojen na ispitu (celobrojna vrednost do 75) i broj poena osvojen na domaćem zadatku (celobrojna vrednost do 25), na osnovu kojih generiše ocenu na predmetu. Ocena može biti: „A“, „B“, „C“ ili „D“. Ocena se računa na osnovu ukupnog broja poena koji se dobija kao zbir poena na ispitu i domaćem zadatku:
 - Ocena „A“ – broj poena veći ili jednak 70
 - Ocena „B“ – broj poena veći ili jednak 50, ali manji od 70
 - Ocena „C“ – broj poena veći ili jednak 30, ali manji od 50
 - Ocena „D“ – broj poena manji od 30
- Ukoliko je ukupan broj poena izvan očekivanog opsega, aplikacija prikazuje poruku „Greška“.



ZADATAK 4 – OCENE 2.0

POSTAVKA (2)

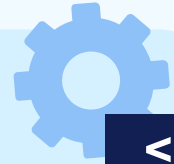


- Marko je testirao svoju aplikaciju korišćenjem metode klasa ekvivalencije. Svi njegovi napisani testovi su uspešno izvršeni, te je on mislio da je njegova aplikacija u potpunosti ispravna. Međutim, aplikacija je Marku izračunala da je iz Testiranja softvera dobio ocenu „B“, a imao je 70 poena, što bi trebalo da bude ocena „A“.
 - On je tada zaključio da ipak nije ispravno implementirao aplikaciju, a takođe da ni testovi koje je napisao nisu bili dovoljno temeljni da otkriju ovaj defekt u aplikaciji. Zbog toga, Marko je napravio verziju 2.0 svoje aplikacije i odlučio da proširi napisani skup test primera, upotrebom metode graničnih vrednosti.
1. Proširiti skup test primera iz zadatka 3 korišćenjem metode graničnih vrednosti.

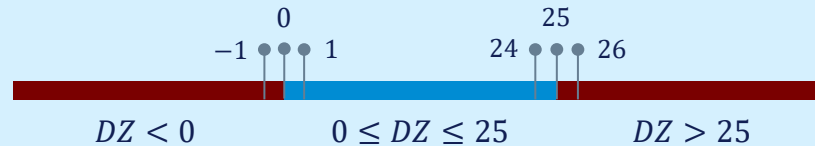
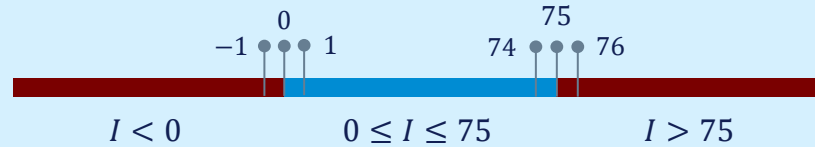


ZADATAK 4 – OCENE 2.0

REŠENJE – KLASJE EKVIVALENCIJE ZA ULAZNE PODATKE



- Klase ekvivalencije za ulazne podatke su iste kao u zadatku 3, dok će same granične vrednosti biti pokrivene u projektovanim testovima.

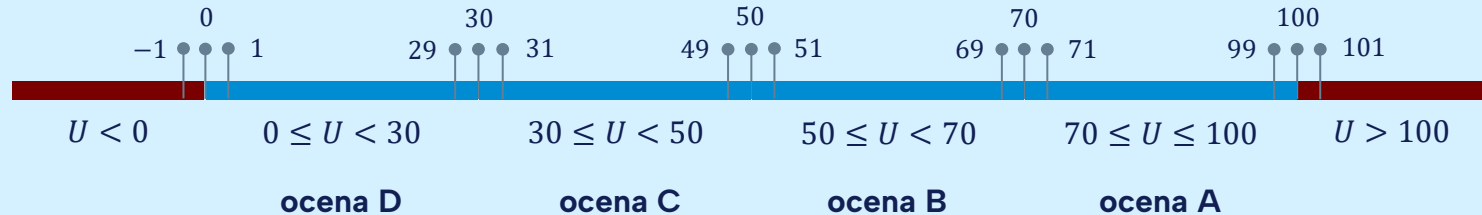


ZADATAK 4 – OCENE 2.0

REŠENJE – KLASJE EKVIVALENCIJE ZA IZLAZNE PODATKE



- Klase ekvivalencije za izlazne podatke su takođe iste kao u zadatku 3, a same granične vrednosti će takođe biti pokrivene u projektovanim testovima.



ZADATAK 4 – OCENE 2.0

REŠENJE – TEST PRIMERI (1)



TP	I	DZ	U	Granična vrednost	Očekivani izlaz	Pokrivene klase ekvivalencije
1	-1	15	14	$I = 0$	Greška	U-2, U-6, I-5
2	0	15	15		Ocena „D“	U-1, U-6, I-4
3	1	15	16		Ocena „D“	U-1, U-6, I-4
4	74	15	89	$I = 75$	Ocena „A“	U-1, U-6, I-1
5	75	15	90		Ocena „A“	U-1, U-6, I-1
6	76	15	91		Greška	U-3, U-6, I-5
7	40	-1	39	$DZ = 0$	Greška	U-1, U-7, I-5
8	40	0	40		Ocena „C“	U-1, U-6, I-3
9	40	1	41		Ocena „C“	U-1, U-6, I-3
10	40	24	64	$DZ = 25$	Ocena „B“	U-1, U-6, I-2
11	40	25	65		Ocena „B“	U-1, U-6, I-2
12	40	26	66		Greška	U-1, U-8, I-5



ZADATAK 4 – OCENE 2.0

REŠENJE – TEST PRIMERI (2)

TP	I	DZ	U	Granična vrednost	Očekivani izlaz	Pokrivene klase ekvivalencije
13	-1	0	-1	$U = 0$	Greška	U-2, U-6, I-5
14	0	0	0		Ocena „D“	U-1, U-6, I-4
15	0	1	1		Ocena „D“	U-1, U-6, I-4
16	29	0	29	$U = 30$	Ocena „D“	U-1, U-6, I-4
17	15	15	30		Ocena „C“	U-1, U-6, I-3
18	6	25	31		Ocena „C“	U-1, U-6, I-3
19	24	25	49	$U = 50$	Ocena „C“	U-1, U-6, I-3
20	50	0	50		Ocena „B“	U-1, U-6, I-2
21	26	25	51		Ocena „B“	U-1, U-6, I-2
22	49	20	69	$U = 70$	Ocena „B“	U-1, U-6, I-2
23	45	25	70		Ocena „A“	U-1, U-6, I-1
24	71	0	71		Ocena „A“	U-1, U-6, I-1
25	74	25	99	$U = 100$	Ocena „A“	U-1, U-6, I-1
26	75	25	100		Ocena „A“	U-1, U-6, I-1
27	75	26	101		Greška	U-1, U-8, I-5



ZADATAK 4 – OCENE 2.0

REŠENJE – TEST PRIMERI (3)

- Da li smo obuhvatili sve realne granice? Šta se dešava sa implementacionim ograničenjima?
 - Treba obratiti pažnju na minimalnu i maksimalnu moguću vrednost tipa podataka u korišćenom programskom jeziku. Za svrhe ovog zadatka, pretpostavićemo da su celi brojevi smešteni na 16 bitova. Dakle, važi: $minint = -32768$ i $maxint = 32767$.

TP	I	DZ	U	Granična vrednost	Očekivani izlaz	Pokrivene klase ekvivalencije
28	32766	15	?	$I = maxint$	Greška	U-3, U-6, I-5
29	32767	15	?		Greška	U-3, U-6, I-5
30	32768	15	?		Greška	U-3, U-6, I-5
31	-32769	15	?	$I = minint$	Greška	U-2, U-6, I-5
32	-32768	15	?		Greška	U-2, U-6, I-5
33	-32767	15	?		Greška	U-2, U-6, I-5

ZADATAK 4 – OCENE 2.0

REŠENJE – TEST PRIMERI (4)



TP	I	DZ	U	Granična vrednost	Očekivani izlaz	Pokrivene klase ekvivalencije
34	40	32766	?	$DZ = maxint$	Greška	U-1, U-8, I-5
35	40	32767	?		Greška	U-1, U-8, I-5
36	40	32768	?		Greška	U-1, U-8, I-5
37	40	-32769	?	$DZ = minint$	Greška	U-1, U-7, I-5
38	40	-32768	?		Greška	U-1, U-7, I-5
39	40	-32767	?		Greška	U-1, U-7, I-5
40	16383	16383	32766	$U = maxint$	Greška	U-3, U-8, I-5
41	32767	0	32767		Greška	U-3, U-6, I-5
42	1	32767	32768		Greška	U-1, U-8, I-5
43	0	-32767	-32767	$U = minint$	Greška	U-1, U-7, I-5
44	-16384	-16384	-32768		Greška	U-2, U-7, I-5
45	-32768	-1	-32769		Greška	U-2, U-7, I-5

PITANJA?

<https://si3ts.etf.bg.ac.rs/>

nikolas@etf.bg.ac.rs

