

1	2	3	4	Σ

- (8 bodova)** Kreirati algoritam kojim se učitava cijeli broj **X** i koji računa njegovu apsolutnu vrijednost. Apsolutna vrijednost **|X|** jednaka je **X** za **X>0**, odnosno **-X** za **X<0**. Na izlazu štampati dobijenu vrijednost.
- (13 bodova)** Kreirati algoritam kojim se učitava string **S** i koji provjerava da li string može predstavljati heksadecimalni zapis boje. Heksadecimalni zapis boje je validan ukoliko počinje karakterom „#“ nakon kojeg slijedi 6 karaktera koji mogu biti cifre ili velika slova iz opsega A-F.
Primjer: String **S="#FBA9C3"** predstavlja validan heksadecimalni zapis boje dok **S="#RM00FF"** ne predstavlja.
- (13 bodova)** Kreirati algoritam kojim se unosi niz cijelih brojeva **X**, dužine **N** i koji formira novi niz **Y** od nenultih, parnih cifara elemenata niza **X**.
Primjer: Ako je **X=[356, 108, 4, 12, 503]** tada je **Y=[6, 8, 4, 2]**.
- (16 bodova)** Kreirati algoritam kojim se unosi matrica **A** čiji su elementi cifre (ne provjeravati), dimenzija **MxN**, i koji od elemenata matrice **A** formira cijeli broj **X**. Matrica se obilazi po vrstama tako da je cifra na poziciji (1, 1) najmanje težine dok je cifra na poziciji (M, N) najveće.

Primjer: Za matricu $A = \begin{bmatrix} 0 & 3 \\ 2 & 7 \end{bmatrix}$ treba formirati broj **X=7230**.

Kolokvijum se radi 50 minuta.

NEČITKO NAPISANI DJELOVI KODA NEĆE BITI PREGLEDANI!

POTREBNO JE CRTATI ALGORITAMSKU ŠEMU. NIJE DOZVOLJENO PISANJE PSEUDOKODA.

1	2	3	4	Σ

- (8 bodova)** Kreirati algoritam kojim se učitava cijeli broj **X** i koja na osnovu vrijednosti **X** računa vrijednost **Y** na sljedeći način: **Y = X** za **X≥0**, odnosno **Y=0** za **X<0**. Na izlazu štampati dobijenu vrijednost.
- (13 bodova)** Kreirati algoritam koji učitava stringove **S1** i **S2** i koji na kraj stringa **S1** nadovezuje samo specijalne karaktere iz stringa **S2**. Specijalni karakteri su svi oni karakteri koji nisu slova ili cifre.
Primjer: String **S1="tekst"** na osnovu **S2="mA\$%k7"** postaje **S1="tekst\$"**.
- (13 bodova)** Kreirati algoritam kojim se unosi niz cijelih brojeva **X**, dužine **N** i koji formira novi niz **Y** od nenultih, neparnih cifara elemenata niza **X**.
Primjer: Ako je **X=[356, 108, 4, 12, 503]** tada je **Y=[3, 5, 1, 1, 5, 3]**.
- (16 bodova)** Kreirati algoritam kojim se unose matrica **A** čiji su elementi cifre (ne provjeravati), dimenzija **MxN**, i koji od elemenata matrice **A** formira cijeli broj **X**. Matrica se obilazi po vrstama tako da je cifra na poziciji (1, 1) najveće težine dok je cifra na poziciji (M, N) najmanje.

Primjer: Za matricu $A = \begin{bmatrix} 0 & 3 \\ 2 & 7 \end{bmatrix}$ treba formirati broj **X=327**.

Kolokvijum se radi 50 minuta.

NEČITKO NAPISANI DJELOVI KODA NEĆE BITI PREGLEDANI!

POTREBNO JE CRTATI ALGORITAMSKU ŠEMU. NIJE DOZVOLJENO PISANJE PSEUDOKODA.

1	2	3	4	Σ

- (8 bodova)** Kreirati algoritam kojim se učitava broj **N** i koji provjerava da li je unijeti broj dvocifren. Ukoliko jeste, na izlazu štampati cifru najmanje težine tog broja, a u suprotnom odštampati poruku „Ne radi se o dvocifrenom broju“.
Pojašnjenje: Cifra najmanje težine broja 27 je 7.
- (13 bodova)** Kreirati algoritam kojim se učitavaju stringovi **S1** i **S2**. Algoritam treba da provjeri u kom stringu je veći zbir cifara (karaktera koji su cifre) i da odštampa odgovarajuću poruku.
Primjer: Za stringove **S1="#FBA9C3"** i **S2="AA34B"**, algoritam treba da odštampa poruku: „Veći je zbor cifara u stringu S1“, jer je $9 + 3 > 3 + 4$.
- (13 bodova)** Kreirati algoritam kojim se unosi niz cijelih brojeva **X**, dužine **N**. Algoritam treba da odredi koliko ima različitih elemenata u nizu X.
Primjer: Za niz **X=[356, 108, 5, 108, 5, 12, 5]**, algoritam treba da odštampa broj 4, jer postoje 4 različita elementa niza X (356, 108, 5 i 12).
- (16 bodova)** Kreirati algoritam kojim se unose matrica **A**, dimenzija **MxN**, i koji formira niz B od maksimalnih elemenata svake vrste.
Primjer: Za matricu $A = \begin{bmatrix} 0 & 3 \\ 2 & 7 \end{bmatrix}$ treba formirati niz **B=[3,7]**.

Kolokvijum se radi 50 minuta.

NEČITKO NAPISANI DJELOVI KODA NEĆE BITI PREGLEDANI!

POTREBNO JE CRTATI ALGORITAMSKU ŠEMU. NIJE DOZVOLJENO PISANJE PSEUDOKODA.

1	2	3	4	Σ

- (8 bodova)** Kreirati algoritam kojim se učitava broj **N** i koji provjerava da li je unijeti broj dvocifren. Ukoliko jeste, na izlazu štampati cifru najveće težine tog broja, a u suprotnom odštampati poruku „Ne radi se o dvocifrenom broju“.
Pojašnjenje: Cifra najveće težine broja 27 je 2.
- (13 bodova)** Kreirati algoritam kojim se učitavaju stringovi **S1** i **S2**. Algoritam treba da provjeri u kom stringu je više slova.
Primjer: Za stringove **S1="#FbA9c3"** i **S2="Aa34B"**, algoritam treba da odštampa poruku: „Više ja slova u stringu S1“, jer se u stringu S1 nalaze 4 slova, a u stringu S2 3 slova.
- (13 bodova)** Kreirati algoritam kojim se unosi niz cijelih brojeva **X**, dužine **N** i koji formira novi niz **Y** od elemenata niza X koji su veći od sebi susjednih elemenata.
Primjer: Ako je **X=[356, 108, 4, 127, 50]** tada je **Y=[356, 127]**.
- (16 bodova)** Kreirati algoritam kojim se unose matrica **A**, dimenzija **MxN**, i koji formira vektor B od maksimalnih elemenata svake kolone.
Primjer: Za matricu $A = \begin{bmatrix} 0 & 3 \\ 2 & 7 \end{bmatrix}$ treba formirati niz **B=[2, 7]**.

Kolokvijum se radi 50 minuta.

NEČITKO NAPISANI DJELOVI KODA NEĆE BITI PREGLEDANI!

POTREBNO JE CRTATI ALGORITAMSKU ŠEMU. NIJE DOZVOLJENO PISANJE PSEUDOKODA.