

Osnovi programiranja (II kolokvijum)

Vreme izrade: 75 minuta

09.08.2025.

Na **Desktop-u**, u direktorijumu **Rad**, kreirati direktorijum **Ime_Prezime_BrojIndeksa_GodinaUpisa** i unutar njega sačuvati program koji sadrži rešenje datog zadatka. Rešenje 1. zadatka **mora** da se nalazi u fajlu **Zadatak1.c**.

NAPOMENA: U zadatku su eksplicitno navedeni potpisi funkcija kojih se treba pridržavati. Ne menjati povratne vrednosti funkcija, nazive, kao ni broj i tipove parametara. Nije dozvoljeno kreiranje dodatnih funkcija.

Zadatak

Kompaniji *Brza Isporka* je potrebna pomoć pri implementaciji osnovnog modula za novi sistem upravljanja magacinom (WMS – *Warehouse Management System*). Magacin je predstavljen kao 2D mreža (matrica), gde svaka ćelija sadrži identifikacioni broj (ID) proizvoda ili je prazna (označena nulom). Potrebno je obraditi listu narudžbina, a nakon toga reorganizovati preostale proizvode kako bi se oslobodio prostor za novu robu.

Napisati program koji sadrži sledeće funkcije:

- **void UcitajMatricu(int m, int n, int matrica[m][n])**, koja učitava elemente matrice celih brojeva dimenzija $m \times n$. Elementi matrice su pozitivni celi brojevi koji predstavljaju ID proizvoda ili 0 (u slučaju da je ćelija prazna); $1 \leq m, n \leq 50$
- **void UcitajNiz(int n, int niz[n])**, koja učitava elemente niza celih brojeva dužine n ($1 \leq n \leq 50$).
- **void StampajStanje(int m, int n, int matrica[m][n])**, koja štampa trenutno stanje magacina (matricu) na standardni izlaz, u matričnom obliku.
- **int UkloniProizvod(int m, int n, int magacin[m][n], int id_proizvoda)**, koja pretražuje matricu *magacin* kako bi pronašla proizvod sa datim *id_proizvoda*. Ukoliko je proizvod pronađen, funkcija ispisuje poruku o lokaciji, menja vrednost te ćelije u 0 (uklanja proizvod) i vraća 1. Poruka treba da bude u formatu:

Proizvod [ID] spakovan sa lokacije: red=[R], kolona=[K],

gde R i K predstavljaju redne brojeve reda i kolone ćelije u kojoj se proizvod nalazio indeksirane od 1.

Funkcija prestaje sa pretragom čim pronađe i ukloni prvi takav proizvod.

Ukoliko proizvod nije pronađen u magacinu, funkcija vraća 0 i ne menja stanje magacina.

- **void ObradiSveNarudzbine(int m, int n, int magacin[m][n], int br_narudzbina, int narudzbine[br_narudzbina])**, koja prolazi kroz niz narudžbine i za svaki ID proizvoda iz narudžbine poziva funkciju *UkloniProizvod*. Ako funkcija *UkloniProizvod* vrati vrednost 0 (što znači da proizvod nije pronađen), ova funkcija ispisuje poruku:

Proizvod [ID] nije na stanju

- **void KonsolidujMagacin(int m, int n, int magacin[m][n])**, koja reorganizuje magacin tako što sve proizvode (elemente različite od nule) premešta na početak matrice, popunjavajući je red po red, s leva na desno. Preostale ćelije matrice se popunjavaju nulama.

NAPOMENA: funkciju *KonsolidujMagacin* napisati tako da se izmena matrice odradi *in-place*, tj. bez alociranja memorije za dodatne strukture (pomoćne matrice ili nizove).

Primer:

$$\begin{bmatrix} 0 & 2 & 0 \\ 3 & 4 & 0 \\ 5 & 7 & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 2 & 3 & 4 \\ 5 & 7 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

U glavnom delu programa potrebno je testirati funkcionalnosti kreiranog sistema na sledeći način:

- Učitati dimenzije magacina i, koristeći funkciju **UcitajMatricu**, učitati početno stanje magacina.
- Učitati broj narudžbina i, koristeći funkciju **UcitajNiz**, učitati ID-jeve naručenih proizvoda.
- Pozvati funkciju **ObradiSveNarudzbine** kako bi se svi proizvodi pronašli i spakovali.
- Ispisati poruku „**Stanje magacina nakon obrade narudžbine:**“ i pozvati **StampajStanje** kako bi se prikazalo stanje magacina.
- Pozvati funkciju **KonsolidujMagacin** kako bi se preostali proizvodi reorganizovali.
- Ispisati poruku „**Stanje magacina nakon konsolidacije:**“ i ponovo pozvati **StampajStanje** kako bi se prikazalo stanje magacina nakon reorganizacije proizvoda.

Test primeri

3 3 1 2 3 4 5 6 7 8 9 1 1	Proizvod [1] spakovan sa lokacije: red=[1], kolona=[1] Stanje magacina nakon obrade narudzbine: 0 2 3 4 5 6 7 8 9 Stanje magacina nakon konsolidacije: 2 3 4 5 6 7 8 9 0
3 3 1 2 3 4 3 1 4 7 8 5 1 2 3 4 5	Proizvod [1] spakovan sa lokacije: red=[1], kolona=[1] Proizvod [2] spakovan sa lokacije: red=[1], kolona=[2] Proizvod [3] spakovan sa lokacije: red=[1], kolona=[3] Proizvod [4] spakovan sa lokacije: red=[2], kolona=[1] Proizvod [5] nije na stanju Stanje magacina nakon obrade narudzbine: 0 0 0 0 3 1 4 7 8 Stanje magacina nakon konsolidacije: 3 1 4 7 8 0 0 0 0
3 5 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 1 10	Proizvod [10] spakovan sa lokacije: red=[2], kolona=[5] Stanje magacina nakon obrade narudzbine: 1 2 3 4 5 6 7 8 9 0 11 12 13 14 15 Stanje magacina nakon konsolidacije: 1 2 3 4 5 6 7 8 9 11 12 13 14 15 0