

Definicija jednodimenzionalnih polja, dodjeljivanje početnih vrijednosti (bez znakovnih polja)

Što smo definirali sljedećom naredbom:

```
int a[8] = {1};
```

Algoritmi s jednodimenzionalnim numeričkim poljima

Napisati program koji računa sumu pozitivnih elemenata u polju od n elemenata.

Algoritmi s jednodimenzionalnim znakovnim poljima

Što će se ispisati sljedećim programom?

```
#include <stdio.h>
#define velpolja 5
int main(){
    char polje[velpolja+1]="ABCDE";
    int i=0;
    while(polje[i]){
        printf("%c%c",polje[i],polje[velpolja-1-i]);
        i++;
    }

    return 0;
}
```

Definicija dvodimenzionalnih i višedimenzionalnih polja i dodjeljivanje početnih vrijednosti dvodimenzionalnim poljima

1. Kako izgleda drugi stupac matrice m definirane na sljedeći način:

```
int m[4][4]={1,2,3,4,5,6};
```

Jednostavniji algoritmi s dvodimenzionalnim poljima

Što će se na zaslonu računala ispisati sljedećim programom?

```
#include <stdio.h>
int main() {
    int a[3][3] = { 6, 5, 3, 4, 5, 6 }, i, j, suma = 0;
    for(i=0;i<3;i++)
        for(j=0;j<3;j++)
            if(i && j) suma+=a[j][2];
    return 0;
}
```

Zauzeće memorije varijabla i poljima (sizeof, ručno brojanje, procjena)

Koliko mjesta u memoriji zauzima sljedeće polje:

```
int a[2][4] = {1, 2, 3};
```

Primjena pokazivača u dvodimenzionalnim poljima

Što će se ispisati izvođenjem sljedećeg programa?

```
#include <stdio.h>
#define MAX 5
int main() {
    int m[MAX][MAX] = {{1, 2}, {3, 4}, {5, 6}};
    int *p = &m[0][0], suma = 0;
    int i, j;

    for (i = 0; i < MAX; i++) {
        for (j = 0; j < MAX; j++) {
            if (*(p + i * MAX + j) % 2 == 0) {
                suma += *(p + i * MAX + j);
            }
        }
    }
    printf ("%d\n", suma);
    return 0;
}
```

Primjena pokazivača u jednodimenzionalnim poljima

Što će se ispisati izvođenjem sljedećeg programa?

```
#include <stdio.h>
int main(){
    int polje[] = {2,4,6,8};
    int *p = &polje[3];
    --p;
    *p = ++*p;
    printf("polje = %d %d %d %d \t*p=%d", polje[0], polje[1],
    polje[2], polje[3], *p);
    return 0;
}
```

Jednodimenzionalna znakovna polja, dodjeljivanje početnih vrijednosti nizu znakova

Ukoliko je sljedeća naredba ispravna, što je s njom definirano.

```
char p[10] = {'a', 'b', 99};
```

Učitavanje polja i ispis polja (samo jednostavni formati kao npr. %5d, %15.7f, %s, %c)

Što radi sljedeći programski odsječak?

```
int polje[10][10], i, j;
for (i=0; i<10; i++)
    for (j=0; j<10; j++)
        polje[i][j] = i;
```

Definiranje pokazivača. Tipovi pokazivača. Korištenje pokazivača (inicijalizacija, pristup podatku, izmjena podatka)

1. Zadatak

Kolika će biti vrijednost pohranjena u varijabli `b` nakon obavljanja sljedećeg programskog odsječka:

```
float *a; float b = 2.78;
a = &b;
*a = 10 * b;
```

2. Zadatak

Koja će biti vrijednost varijabli `x` i `pp` nakon izvođenja sljedećeg programskog odsječka ?

```
#include <stdio.h>
int main() {
    int x = 5, *pp;
    pp = &x;
    *pp = 10;
    return 0;
}
```

Korištenje jednodimenzionalnih polja (pristupanje članovima polja, indeksni izrazi)

Što će ispisati sljedeći programski odsječak?

```
#include <stdio.h>
int main()
{
    int polje [10] = {10, 9, 8, 7, 6, 5, 4, 3, 2, 1}, i;
    i = 0;
    while(i<10)
    {
        polje[i/2] /= 2;
        i++;
    }
    for(i=0; i<10; i++)
    {
        printf("%d ", polje[i]);
    }
    return 0;
}
```

RJEŠENJA ZADATAKA

Definicija jednodimenzionalnih polja, dodjeljivanje početnih vrijednosti (bez znakovnih polja)

Rješenje: Polje cjelobrojnih varijabli od 8 elemenata, gdje je prvi element polja a[0], a zadnji a[7], te je vrijednost prvog elementa 1 a ostalih 0.

Algoritmi s jednodimenzionalnim numeričkim poljima

Rješenje:

```
int main(){
    int suma = 0, i, n;
    //...
    for (i = 0; i < n; i++)
        if (a[i] >= 0) suma += a[i];
    return 0;
}
```

Algoritmi s jednodimenzionalnim znakovnim poljima

Rješenje:

AEBDCCDBEA

Definicija dvodimenzionalnih i višedimenzionalnih polja i dodjeljivanje početnih vrijednosti dvodimenzionalnim poljima

Rješenje:

2
6
0
0

Jednostavniji algoritmi s dvodimenzionalnim poljima

Rješenje: 12

Zauzeće memorije varijablama i poljima (sizeof, ručno brojanje, procjena)

Rješenje: 32

Primjena pokazivača u dvodimenzionalnim poljima

Rješenje: 12

Primjena pokazivača u jednodimenzionalnim poljima

Rješenje: polje = 2 4 7 8 *p=7

Jednodimenzionalna znakovna polja, dodjeljivanje početnih vrijednosti nizu znakova

Rješenje: Naredba je ispravna i njome je definirano polje znakova veličine 10. Prvi znak u polju (tj. $p[0]$) je znak 'a', sljedeći znak je znak 'b', a na mjestu $p[2]$ u polju se nalazi znak čiji je ASCII kod 99, a to je znak 'c'. Preostali znakovi u polju (dakle, od $p[3]$ do $p[9]$) imaju ASCII vrijednost 0.

Učitavanje polja i ispis polja (samo jednostavni formati kao npr. %5d, %15.7f, %s, %c)

Rješenje: Vrijednost svakog člana polja postavi na indeks retka u kojem se član nalazi.

Definiranje pokazivača. Tipovi pokazivača. Korištenje pokazivača (inicijalizacija, pristup podatku, izmjena podatka)

1. Zadatak

Rješenje: 27.8

2. Zadatak

Rješenje: $x=10$, pp je adresa memorijske lokacije varijable x

Korištenje jednodimenzionalnih polja (pristupanje članovima polja, indeksni izrazi)

2 2 2 1 1 5 4 3 2 1

Kratko objašnjenje: Indeksni izraz kojim se mijenja sadržaj polja ($[i/2]$) utječe samo na prvu polovicu polja i to na svaki element po dva puta (preciznije, svaki element iz prve polovice polja dijeli po dva puta). Kod ovakvih primjera treba paziti iz razloga što iako petlja ide do gornje granice polja (u primjeru je to 10) ne mijenja se svaki element polja – to ovisi o indeksnom izrazu!

ZADACI

Definicija jednodimenzionalnih polja, dodjeljivanje početnih vrijednosti (bez znakovnih polja)

1. zadatak

Koje su vrijednosti pohranjene u elementima polja ako je polje definirano i inicijalizirano na sljedeći način:

```
int p[5]={1};
```

2. zadatak

Koje su vrijednosti pohranjene u elementima polja ako je polje definirano na sljedeći način:

```
int p[5];
```

Algoritmi s jednodimenzionalnim numeričkim poljima

1. zadatak

Napisati odsječak kôda koji ispisuje indekse onih elemenata polja od n elemenata koji nisu djeljivi sa 2.

2. zadatak

Napisati odsječak kôda koji računa umnožak elemenata polja od n elemenata.

Algoritmi s jednodimenzionalnim znakovnim poljima

1. zadatak

Napisati program koji pri ispisu "okrene" prvu polovicu nekog polja znakova, a drugu polovicu ispiše u poretku kako su znakovi i pohranjeni u polju. Npr. za polje "ABCDEFGH I " treba ispisati EDCBAFGH I . Testirati rad programa za paran i neparan broj elemenata polja.

2. zadatak

Što se ispiše sljedećim programom?

```
#include <stdio.h>
int main(){
    char polje[]="qwertzuiop";
    char tmp;
    int i;
    for(i=0;i<5;i++){
        if(polje[i]<polje[9-i]){
            tmp=polje[i];
            polje[i]=polje[9-i];
            polje[9-i]=tmp;
        }
    }
    for(i=0;i<10;i++){
        printf("%c",polje[i]);
    }
    return 0;
}
```

Definicija dvodimenzionalnih i višedimenzionalnih polja i dodjeljivanje početnih vrijednosti dvodimenzionalnim poljima

1. zadatak

Kako izgleda drugi stupac matrice definirane na sljedeći na način:

```
int m[4][4] = { {1, 2, 3}, {4, 5, 6} };
```

2. zadatak

Što je neispravno u sljedećoj definiciji?

```
int m[4][0][4] = {1, 2, 3, 4, 5, 6};
```

Jednostavniji algoritmi s dvodimenzionalnim poljima

1. zadatak

Napisati program koji će učitati matricu dimenzija 8*8 i ispisati sumu elemenata koji se ne nalaze na "obodu" (obod definiramo kao uniju elemenata prvog retka, prvog stupca, zadnjeg retka i zadnjeg stupca) matrice.

2. zadatak

Što će se ispisati na zaslonu računala sljedećim programskim odsječkom?

```
int m[2][3] = {1, 2, 3, 4, 5, 6};
int i, j, suma = 0;
for (i = 0; i < 2; i++)
    for (j = 0; j <= i; j++)
        suma += m[i][j];
printf ("%d", suma);
```

Zauzeće memorije varijablama i poljima (sizeof, ručno brojanje, procjena)

1. zadatak

Koliko je bajtova rezervirano u memoriji sljedećom definicijom:

```
char p[4][3][2];
```

2. zadatak

Koliko će bajtova biti utrošeno za pohranu u memoriji polja a?

```
int a[5], i;
for (i = 1; i <= 5; i++)
    a[i] = (float) i / 2;
```

Primjena pokazivača u dvodimenzionalnim poljima

1. zadatak

Učitajte vrijednosti za broj redaka (brRedaka) i broj stupaca (brStupaca) dvodimenzionalnog realnog polja, koje moraju biti manje ili jednake 10, a zatim učitajte članove tog polja.

Potrebno je izračunati i ispisati umnožak svih članova matrice različitih od 0. Zadatak riješiti korištenjem pokazivača na dvodimenzionalno polje.

2. zadatak

Što će se ispisati izvođenjem sljedećeg programa?

```
#include <stdio.h>
#define MAX 5
int main() {
    float m[MAX][MAX] = {{1, 2}, {3, 4}};
    float *p = &m[0][0], umnozak = 1.;
    int i, j;

    for (i = 0; i < MAX; i++) {
        if (*(p + i * MAX + i) != 0) {
            umnozak *= *(p + i * MAX + i);
        }
    }
    printf ("%f\n", umnozak);
    return 0;
}
```

Primjena pokazivača u jednodimenzionalnim poljima

1. zadatak

Učitajte vrijednost za broj elemenata (brElem) jednodimenzionalnog realnog polja, koje mora biti manje ili jednake 10, a zatim učitajte članove tog polja.

Potrebno je izračunati i ispisati omjer najvećeg i najmanjeg člana.

Npr. za polje 1, 7, -2, 9 omjer najvećeg i najmanjeg člana je $9/-2 = -4.5$.

Zadatak riješiti korištenjem pokazivača na jednodimenzionalno polje.

2. zadatak

Što će se ispisati izvođenjem sljedećeg programa?

```
#include <stdio.h>

int main(){

    int polje[] = {2,4,6,8};
    int *p = &polje[3];
    p -= 2;
    *p += *p;
    p++;
    printf("polje = %d %d %d %d \t*p=%d", polje[0], polje[1],
polje[2], polje[3], *p);
    return 0;
}
```


Učitavanje polja i ispis polja (samo jednostavni formati kao npr. %5d, %15.7f, %s, %c)

1. Zadatak

Neka je p kvadratna matrica dimenzija N x N. Što radi sljedeći programski odsječak?

```
for(j=0; j<N; j++) {  
    for(i=0; i<N; i++)  
        printf("%d ",p[i][j]);  
    printf("\n");  
}
```

2. Zadatak

Napisati programski odsječak kojim se učitavaju elementi na glavnoj dijagonali matrice deklarirane kao float a[M][M]?

Definiranje pokazivača. Tipovi pokazivača. Korištenje pokazivača (inicijalizacija, pristup podatku, izmjena podatka)

1. Zadatak

Pažljivo proučite program:

```
#include <stdio.h>  
int main ( ) {  
    int a = 3, *b;  
    b = &a;  
    scanf("%d", b);  
    printf("%d\n", a);  
    return 0;  
}
```

i odgovorite što će se ispisati na ekran ako se na tipkovnici utipka 9 i Enter?

2. Zadatak

Što će se ispisati sljedećim programom?

```
#include <stdio.h>  
int main() {  
    int a = 2, b, *p;  
    p = &a;  
    b = *p;  
    printf("a=%d b=%d\n", a, b);  
    return 0;  
}
```

Korištenje jednodimenzionalnih polja (pristupanje članovima polja, indeksni izrazi)

1. Zadatak

Što će se ispisati sljedećim programskim odsječkom?

```
#include <stdio.h>

int main()
{
    int polje [10] = {7, 6, 3, 4, 10, 9, 1, 5, 2, 8}, i;
    i = 0;
    do
    {
        if(i%2 && polje[i]%2) printf("%d", polje[i]);
        i++;
    } while(i<10);
}
```

2. Zadatak

Programer kojega smo predstavili na prethodnoj provjeri znanja na računalu i koji nam je zadao glavobolju svojom brzopletošću se vratio – izradio je programski odsječak kojim je treba ispisati cijele brojeve uzlazno pohranjene u polju silaznim redoslijedom.

```
#include <stdio.h>

int main()
{
    int polje [10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}, i, j, k;
    i = 0;
    for(i = 0; i<10; i++)
    {
        j = 10-i;
        printf("%d ", polje[j]);
    }
}
```

Iako je program na prvo pogled dobar, postoji pogreška koja je vidljiva po pokretanju programa:

-858993460 10 9 8 7 6 5 4 3 2

Vidljivo je da će programer trebati pomoć FER-ovaca: pomognimo mu i pronađimo gdje je pogriješio!

Jednodimenzionalna znakovna polja, dodjeljivanje početnih vrijednosti nizu znakova

1. Zadatak

Koje od sljedećih naredbi nisu ispravne:

- 1) `char p[] = { 'A' , 65 , 'C' } ;`
- 2) `char p[2] = { 'A' , 0 , 0 } ;`
- 3) `char p[4] = { 'A' , 0 , 0 } ;`
- 4) `char p[10] = { 1 , 2 , 3 } ;`
- 5) `char p[2] = { "AB" , "CD" } ;`
- 6) `char p[4] = { "AB" , "CD" } ;`

2. Zadatak

Ukoliko je polje p definirano kao:

```
char p[3];
```

koje od sljedećih naredbi nisu ispravne:

- 1) `p['A'] = 'C' ;`
- 2) `p['3'] = 'C' ;`
- 3) `p['2'] = 'C' ;`
- 4) `p[1] = 'C'`
- 5) `p[3] = 67`
- 6) `p[2] = 32;`

RJEŠENJA

Definicija jednodimenzionalnih polja, dodjeljivanje početnih vrijednosti (bez znakovnih polja)

1. zadatak

1 0 0 0 0

2. zadatak

nepoznate vrijednosti (ne možemo točno reći koje)

Algoritmi s jednodimenzionalnim numeričkim poljima

1. zadatak

```
int suma = 0, i, n;
//...
for (i = 0; i < n; i++)
    if (a[i] % 2) printf(„%d“, i);
```

2. zadatak

```
int umnozak = 1, i, n;
//...
for (i = 0; i < n; i++)
    umnozak *= a[i];
```

Algoritmi s jednodimenzionalnim znakovnim poljima

1. zadatak

```
#include <stdio.h>
#define velpolja 8
int main(){
    char polje[velpolja+1]="ABCDEFGH";
    int i=0;
    for(i=0;i<velpolja;i++){
        if(i<=(velpolja-1)/2)
            printf("%c",polje[(velpolja-1)/2-i]);
        else
            printf("%c",polje[i]);
    }
    return 0;
}
```

2. zadatak

qwiuztreop

Definicija dvodimenzionalnih i višedimenzionalnih polja i dodjeljivanje početnih vrijednosti dvodimenzionalnim poljima

1. zadatak

2
5
0
0

2. zadatak

Ni jedna dimenzija kod višedimenzionalnih polja ne smije biti manja od 1!

Jednostavniji algoritmi s dvodimenzionalnim poljima

1. zadatak

```
#include <stdio.h>
#define BR_RED 8
#define BR_STUP 8

int main() {
    int mat[BR_RED][BR_STUP];
    int i, j, suma = 0;
    for(i=0; i<BR_RED; i++){
        for(j=0; j<BR_STUP; j++){
            scanf("%d", &mat[i][j]);
            if (i!=0 && j!=0 && i!=BR_RED-1 && j!=BR_STUP-1){
                suma += mat[i][j];
            }
        }
    }
    printf ("%d", suma);
    return 0;
}
```

2. zadatak

10

Zauzeće memorije varijablama i poljima (sizeof, ručno brojanje, procjena)

1. zadatak

24

2. zadatak

20

Primjena pokazivača u dvodimenzionalnim poljima

1. zadatak

```
#include <stdio.h>
#define MAXRED 10
#define MAXSTUP 10
int main() {
    float m[MAXRED][MAXSTUP];
    float *p = &m[0][0], umnozak = 1.;
    int brRedaka, brStupaca, i, j;

    printf("Upisite broj redaka i broj stupaca: ");
    scanf("%d %d", &brRedaka, &brStupaca);

    // upis clanova polja
    for (i = 0; i < brRedaka; i++) {
        for (j = 0; j < brStupaca; j++) {
            scanf("%f", p + i * MAXSTUP + j);
        }
    }

    // umnozak svih clanova polja
    for (i = 0; i < brRedaka; i++) {
        for (j = 0; j < brStupaca; j++) {
            if (*(p + i * MAXSTUP + j) != 0) {
                umnozak *= *(p + i * MAXSTUP + j);
            }
        }
    }
    printf ("%f\n", umnozak);
    return 0;
}
```

2. zadatak

4.000000

Primjena pokazivača u jednodimenzionalnim poljima

1. zadatak

```
#include <stdio.h>

#define MAX 10

int main() {
    float polje[MAX];
    float *p = &polje[0];
    float max, min;
    int brElem, i;
```

```

do {
    printf("Upisite broj elemenata: ");
    scanf("%d", &brElem);
} while (brElem < 1 || brElem > 10);

// upis clanova polja
for (i = 0; i < brElem; i++) {
    scanf("%f", p + i);
}

min = max = *p;
for (i = 1; i < brElem; i++) { /* primijetiti da petlja
kreće od 1 */
    if (min > *(p + i)){
        min = *(p + i);
    } else if (max < *(p + i)){
        max = *(p + i);
    }

}
/* Sto se događa ako je min=0? */
printf ("Omjer je %f/%f=%f", max, min, max/min);
return 0;
}

```

2. zadatak

polje = 2 8 6 8 *p=6

Učitavanje polja i ispis polja (samo jednostavni formati kao npr. %5d, %15.7f, %s, %c)

1. Zadatak

Rješenje: ispisuje matricu koja nastaje transponiranjem matrice p.

2. Zadatak

Rješenje: for (i = 0; i < M; i++) scanf("%f", &a[i][i]);

Definiranje pokazivača. Tipovi pokazivača. Korištenje pokazivača (inicijalizacija, pristup podatku, izmjena podatka)

1. Zadatak

Rješenje: 9

2. Zadatak

Rješenje: a=2 b=2

Korištenje jednodimenzionalnih polja (pristupanje članovima polja, indeksni izrazi)

1. Zadatak

95

Kratko objašnjenje: Programskim se odsječkom ispisuje sadržaj polja na način da se ispisuje samo element koji se nalazi na neparnoj poziciji u polju (indeks polja za taj element je neparan broj) i koji istovremeno ima neparnu vrijednost. Pažnja: razlikovati indekse elemenata (tj. poziciju unutar polja) i vrijednost elemenata; da bi saznali vrijednost nekog elemenata polja nužno moramo znati njegovu poziciju u polju, tj. indeks.

2. Zadatak

Programer je načinio jednu od klasičnih grešaka (koja je dosta teško uočljiva i s kojom se programeri hrvaju dan za danom), a ta je da je koristio nedozvoljene indekse polja: krenuo je od $j=10$ do $j=1$. Naravno, indeksi polja idu od 0 do 9 te je zbog toga njegov program dao "čudan" prvi broj – otišao je u memoriju "iza" elementa s indeksom 9 i pročitao "smeće" kao element s indeksom 10 (nešto što se posve slučajno našlo u memoriji u tome trenu). Da bi program ispravno radio potrebno je načiniti sljedeći ispravak:

```
j = 10-i-1;
```

Alternativno (ponešto i elegantnije) može se koristiti i indeksni izraz koji u sebi sadrži varijablu unutar for petlje:

```
for(i = 0; i<10; i++)  
{  
    printf("%d ", polje[10-i-1]);  
}
```

Jednodimenzionalna znakovna polja, dodjeljivanje početnih vrijednosti nizu znakova

1. Zadatak

Rješenje: 2 5 6

2. Zadatak

Rješenje: 1 2 3 5