

Redni broj pitanja

< **1** >

Odgovor

Točan  
odgovor

Broj mogućih točnih odgovora: 1

Mogući broj bodova: 0,50

Ostvareni broj bodova: 0,50

1 2 3 4 5 6 7 8 9 10

c a a e e b d b a e

c a a e e b d a a e



Kraj pregledavanja

Što će ispisati sljedeći program?

```
void pisi3(int *p, int i, int n) {
    if (i < n/2) {
        pisi3(p, i+1, n);
        printf ("%d", *(p+i));
        printf ("%d", *(p+n-i-1));
    }
}

int main() {
    int p[4]={1,2,3,4};
    pisi3 (p,0,4);
    return 1;
}
```

a) 1234

b) 4321

☒ c) 2314

d) 3241

e) 1423

Redni broj pitanja

< 2 >

Broj mogućih tačnih odgovora: 1

Mogući broj bodova: 0,50

Ostvareni broj bodova: 0,50

Odgovor

Točan  
odgovor

1 2 3 4 5 6 7 8 9 10

c a a e e b d b a e

c a a e e b d a a e



Kraj pregledavanja

Funkcija za dodavanje elementa na stoga realiziran listom glasi:

a)

```
int dodaj (int element, Stog *stog) {  
    atom *novi;  
    if ((novi = (atom*) malloc(sizeof(atom))) != NULL) {  
        novi->element = element;  
        novi->sljed = stog->vrh;  
        stog->vrh = novi;  
        return 1;  
    }  
    else return 0;  
}
```

b)

```
int dodaj (int element, Stog *stog) {  
    atom *novi;  
    if ((novi = (atom*) malloc(sizeof(atom))) != NULL) {  
        novi = element;  
        return 1;  
    }  
    else return 0;  
}
```

```
int dodaj (int element, Stog *stog) {  
    atom *novi;  
    if ((novi = (atom*) malloc(sizeof(atom))) != NULL) {  
        novi->element = stog->vrh;  
    }
```

Redni broj pitanja

< **3** >

Odgovor

Točan  
odgovor

Broj mogućih točnih odgovora: 1

Mogući broj bodova: 0,50

Ostvareni broj bodova: 0,50

1 2 3 4 5 6 7 8 9 10

c a a e e b d b a e

c a a e e b d a a e



Kraj pregledavanja

Rekurzivna funkcija za izračunavanje n-tog Fibonnacijevog broja proširena je jednom naredbom za ispis te je njen kôd:

```
int F(int n) {  
    if (n <= 1)  
        return 1;  
    else {  
        printf("Test\n");  
        return F(n-2) + F(n-1);  
    }  
}
```

Koliko puta će se na ekranu ispisati riječ "Test", ako se funkcija pozove s F(4)

☒ a) 4

☐ b) 3

☐ c) 1

☐ d) 7

☐ e) 9

Redni broj pitanja

< **4** >

Odgovor

Točan  
odgovor

1 2 3 4 5 6 7 8 9 10

c a a e e b d b a e

c a a e e b d a a e



Kraj pregledavanja

Broj mogućih točnih odgovora: 1

Mogući broj bodova: 0,50

Ostvareni broj bodova: 0,50

Neka je zadana sljedeća rekurzivna funkcija:

```
int F(n) {  
    if (n > 1)  
        return F(n-2) + F(n-4) + F(n-6);  
    else if (n==1)  
        return 1;  
    else  
        return 0;  
}
```

Koju vrijednost će funkcija vratiti u glavni program, ako je poziv funkcije bio F(4)

a) 4

b) 1

c) 3

d) 7

☒ e) 0

Redni broj pitanja

< **5** >

Odgovor

Točan  
odgovor

Broj mogućih točnih odgovora: 1

Mogući broj bodova: 0,50

Ostvareni broj bodova: 0,50

1 2 3 4 5 6 7 8 9 10

c a a e e b d b a e

c a a e e b d a a e



Kraj pregledavanja

Složenost Mergesorta je:

a)  $O(n)$

b)  $O(\log_2 n)$

c)  $O(n^{3/2})$

d)  $O(n^2)$

☒ e)  $O(n \log_2 n)$



Redni broj pitanja

< **6** >

Odgovor

Točan  
odgovor

Broj mogućih točnih odgovora: 1

Mogući broj bodova: 0,50

Ostvareni broj bodova: 0,50

1	2	3	4	5	6	7	8	9	10
c	a	a	e	e	b	d	b	a	e
c	a	a	e	e	b	d	a	a	e
👍	👍	👍	👍	👍	👍	👍	👎	👍	👍

Kraj pregledavanja

Između 5 ponuđenih algoritama za uzlazno sortiranje niza

1, 3, 2, 5, 4, 7, 6, ...,

999997, 999996, 999999, 999998, 1000000

najbolje je odabrati:

a) Shell sort sa Sedgwickovim slijedom: {1, 5, 19, 41, 109, ...}

**b)** Poboljšani bubble sort

c) Selection sort

d) Shell sort s Hibbardovim slijedom: {1, 3, 7, ...,  $2^k - 1$ }

e) Mergesort

Redni broj pitanja

< **7** >

Odgovor

Točan  
odgovor

Broj mogućih točnih odgovora: 1

Mogući broj bodova: 0,50

Ostvareni broj bodova: 0,50

1 2 3 4 5 6 7 8 9 10

c a a e e b d b a e

c a a e e b d a a e



Kraj pregledavanja

Funkcija `push` stavlja elemente na stog. Ako je operacija uspješno obavljena funkcija vraća 1, a u slučaju greške vraća 0. Prototip funkcije je:

```
int push (int element, Stog *stog);
```

Funkcija `pop` skida element sa stoga i vraća njegovu vrijednost ili -1 u slučaju greške. Prototip funkcije `pop` je:

```
int pop (Stog *stog);
```

Što će ispisati sljedeći programski odsječak, uz pretpostavku da je prije izvođenja stog prazan i da na njemu ima dovoljno mjesta.

```
for (i=0; i<5; i++)  
    push(i, &stog);  
  
for (i=5; i>=0; i--)  
    printf("%d ", pop(&stog));
```

- a) 0 1 2 3 4
- b) 5 4 3 2 1 0
- c) 4 3 2 1 0
- ☒ d) 4 3 2 1 0 -1
- e) 0 1 2 3 4 5

Redni broj pitanja

< **8** >

Odgovor

Točan  
odgovor

1	2	3	4	5	6	7	8	9	10
c	a	a	e	e	b	d	b	a	e
c	a	a	e	e	b	d	a	a	e
😊	😊	😊	😊	😊	😊	😊	😞	😊	😊

Kraj pregledavanja

Broj mogućih točnih odgovora: 1

Mogući broj bodova: 0,50

Ostvareni broj bodova: -0,10

Između 5 ponuđenih algoritama za uzlazno sortiranje niza:

100000, 99999, 99998, 99997, 99996, 99995, ...  
15, 14, 13, 12, 11

najbolje je odabrati:

- ☐ a) Merge sort
- ☒ b) Shell sort sa Sedgwickovim slijedom: {1, 5, 19, 41, 109, ...}
- ☐ c) Poboljšani bubble sort
- ☐ d) Shell sort s Hibbardovim slijedom: {1, 3, 7, ...,  $2k-1$ }
- ☐ e) Selection sort



Redni broj pitanja

< **9** >

Odgovor

Točan  
odgovor

1 2 3 4 5 6 7 8 9 10

c a a e e b d b a e

c a a e e b d a a e



Kraj pregledavanja

Broj mogućih točnih odgovora: 1

Mogući broj bodova: 0,50

Ostvareni broj bodova: 0,50

Ulazni niz **7 6 9 4 1 10 8 2 3 5** se sortira Quick sortom pri čemu se stožer bira procjenom medijana na temelju 3 elementa(prvi,zadnji i element na polovici). Kako izgleda niz nakon što je stožer stavljen na predzadnje mjesto?

a) **1 6 9 4 3 10 8 2 5 7**

b) **7 6 9 4 10 8 2 3 1 5**

c) **7 6 9 4 3 10 8 2 1 5**

d) **5 6 9 4 10 8 2 3 1 7**

e) **1 6 9 4 10 8 2 3 5 7**

Redni broj pitanja

< **10** >

Odgovor

Točan  
odgovor

Broj mogućih točnih odgovora: 1

Mogući broj bodova: 0,50

Ostvareni broj bodova: 0,50

1 2 3 4 5 6 7 8 9 10

c a a e e b d b a e

c a a e e b d a a e



Kraj pregledavanja

Koji od sljedećih nizova vodi najgorem izboru stožera u prvom koraku Quick sorta procjenom medijana na temelju 3 elementa(prvi, zadnji i element na sredini)?

a) 9 8 7 6 5 4 3 2 1

b) 5 4 9 6 1 2 7 8 3

c) 4 5 6 1 7 3 2 8 9

d) 9 7 2 4 6 1 5 8 3

**e)** 6 5 7 2 9 3 1 4 8