

მონაცემთა სტრუქტურები და ალგორითმები (I-II კურსი)

II სემესტრი.2020-2021 წელი

(დასკვნითი გამოცდის ნიმუში)

კითხვა 1

კითხვის ტექსტი

დახარისხების რომელი ხერხია გამოყენებული მოცემულ პროგრამაში

```
int main ()
{
    const int arraySize = 10;
    int A[arraySize] = {2, 6, 4, 8, 19, 100, 89, 68, 45, 37};
    int hold;
    cout << მასივის ელემენტები საწყისი თანმიმდევრობით:" << "\n ";
    for (int i = 0; i < arraySize; i++)
        cout << A [i] << " ";

    for(int pass = 1; pass < arraySize; pass++)
        for (int i = 0; i < arraySize-pass; i++)
            if (A[i] > A[i + 1]) {
                hold = A[i];
                A[i] = A[i + 1];
                A[i + 1] = hold;
            }
    cout << " \n დახარისხებული მასივის ელემენტები \n ";
    for (int i = 0; i < arraySize; i++)
        cout << A[i] << " ";
    cout << " \n ";
    getch();
    return 0;
}
```

აირჩიეთ ერთი:

- ☐ a. დახარისხება სელექციის მეთოდით
- ☐ b. კომბინირებული ალგორითმი
- ☐ c. არც ერთი
- ☐ d. ბუშისებრი-ჩაძირვის ალგორითმი

კითხვა 2

რას აკეთებს პროგრამის შემდეგი ფრაგმენტი?

```
for (i=0;i<4;i++)
{
```

```
for (pass = 1; pass < 3; pass++)
```

```
    for (j = 0; j < 2; j++)
```

```
        if (A[i][j] > A[i][j+1])
```

```
        {
```

```
            hold = A[i][j];
```

```
            A[i][j] = A[i][j+1];
```

```
            A[i][j+1] = hold;
```

```
        }
```

აირჩიეთ ერთი:

- ☐ a. მატრიცის თითოეული სტრიქონის ელემენტებს ალაგებს კლებადობის მიხედვით
- ☐ b. მატრიცის თითოეულ სვეტის ელემენტებს ალაგებს ზრდადობით
- ☐ c. მატრიცის თითოეული სტრიქონის ელემენტებს ალაგებს ზრდადობის მიხედვით
- ☐ d. მატრიცის თითოეულ სვეტის ელემენტებს ალაგებს კლებადობით

კითხვა 3

1024 ელემენტიან დახარისხებულ მასივში ორობითი ძებნის დროს საძიებელი სიდიდის მოსაძებნად საჭიროა

აირჩიეთ ერთი:

- ☐ a. საძიებელ სიდიდესთან არანაკლებ 1024 შედარება
- ☐ b. საძიებელ სიდიდესთან არაუმეტეს 10 შედარება
- ☐ c. საძიებელ სიდიდესთან არაუმეტეს 20 შედარება
- ☐ d. საძიებელ სიდიდესთან არაუმეტეს 512 შედარება

კითხვა 4

მოცემულია პროგრამა:

```
int main()
```

```
{
```

```
const int arraySize = 10;
```

```
int A[arraySize] = {2, 4, 6, 8, 19, 37, 45, 68, 89, 100};
```

```
int i, low = 0, high = arraySize, middle, searchKey, iKey,
```

```
count = 0;
```

```
    cout << " შემოიტანეთ საძიებელი მნიშვნელობა \n ";
```

```
    cin >> searchKey;
```

```
    while (low <= high) {
```

```
        middle = (low + high) / 2;
```

```
        if (searchKey == A[middle]) {
```

```
            iKey = middle;
```

```
            count++;
```

```
            break;
```

```
        }
```

```
        if (searchKey < A[middle])
```

```
            high = middle - 1; /* მასივის ზედა ზღვარი */
```

```

else if (searchKey > A[middle])
    low = middle + 1;    /* მასივის ქვედა ზღვარი
*/

    }

if (count != 0)
cout << " მასივში მოცემული მნიშვნელობის რიგითი ნომერია - " <<
iKey;
else
    cout << " მასივში საძიებელი მნიშვნელობის ტოლი ელემენტი არ არის ! \n ";
getch();
return 0;
}

```

ციკლის რომელ ბიჯზე მოხდება საძიებელი მნიშვნელობის რიგითი ნომრის მოძებნა თუ searchKey ცვლადის მნიშვნელობას შეიტანო 37-ს

აირჩიეთ ერთი:

- ☐ a. პირველ ბიჯზე
- ☐ b. მე- 2 ბიჯზე
- ☐ c. მე-3 ბიჯზე
- ☐ d. მე- 5 ბიჯზე

კითხვა 5

ჩამოთვლილთაგან რომელია strcpy ფუნქციის პარამეტრები?

აირჩიეთ ერთი:

- ☐ a. სტრიქონი და მთელი რიცხვი;
- ☐ b. ორი სტრიქონი.
- ☐ c. ორი მთელი რიცხვი;

კითხვა 6

რა მოქმედებას ასრულებენ ფუნქციები strcpy (s1,s2) და strcpy_s (s1,s2) ?

აირჩიეთ ერთი:

- ☐ a. გადააბამს s1 და s2 სტრიქონებს ერთმანეთს და დააბრუნებს მას ფუნქციის მნიშვნელობად
- ☐ b. s2 სტრიქონის კოპირებას s1-ში
- ☐ c. შეადარებს s1 და s2 სტრიქონებს ერთმანეთს
- ☐ d. s1-ში ჩაწერს s2 სტრიქონის სიგრძეს

კითხვა 7

რა მოქმედებას ასრულებენ ფუნქციები strcat (s1,s2) და strcat_s (s1,s2) ?

აირჩიეთ ერთი:

- ☐ a. s2 სტრიქონის კოპირებას s1-ში
- ☐ b. s1 სტრიქონიდან ამოჭრის სიმბოლოთა s2 მიმდევრობას
- ☐ c. s1 სტრიქონიდან ამოჭრის s2-ში შემავალ სიმბოლოებს
- ☐ d. s1 სტრიქონს მიუერთებს s2-ს

კითხვა 8

რა მოქმედებას ასრულებენ ფუნქციები strcmp (s1,s2) და strcmp_s (s1,s2) ?

აირჩიეთ ერთი:

- ☐ a. აღარებს ერთმანეთს s1 და s2 სტრიქონებს
- ☐ b. s1 სტრიქონზე ბოლოში მიაბამს s2-ს
- ☐ c. s2 სტრიქონის კოპირებას s1-ში
- ☐ d. s1 სტრიქონიდან ამოჭრის სიმბოლოთა s2 მიმდევრობას

კითხვა 9

რომელი თავის ფაილის ჩართვაა საჭირო პროგრამაში სტრიქონების დამუშავების ბიბლიოთეკური ფუნქციების (strcpy (), strcat ()) და სხვა) გამოსაყენებლად?

აირჩიეთ ერთი:

- ☐ a. stdlib
- ☐ b. cstring
- ☐ c. iostream
- ☐ d. iomanip

კითხვა 10

რა დაიბეჭდება მოცემული ფრაგმენტის შესრულების შედეგად:

```
char S1[10]="ABC";  
char S2[10]="abc";  
cout<<strcmp(S1,S2);
```

აირჩიეთ ერთი:

- ☐ a. 1
- ☐ b. 0
- ☐ c. -1

კითხვა 11

რა დაიბეჭდება მოცემული ფრაგმენტის შესრულების შედეგად:

```
char b[20]="information";  
cout<<strlen(b);
```

Answer:

კითხვა 12

რა დაიბეჭდება მოცემული კოდის შესრულების შედეგად:

```
int main()
{
char S1[10]={'a','b','c','d','e','f'};

for(int i=0;S1[i];i++)

S1[i]=toupper(S1[i]);

cout<<S1;
_getch();
return 0;
}
```

Answer:

კითხვა 13

რას შეასრულებს შემდეგი ფრაგმენტი?

```
const int N=10;
int a[N], i;
for (i=0; i<N; i++)
    a[i]=rand()%31;
```

აირჩიეთ ერთი:

- ☐ a. 0-დან 31-ჩათვლით ყველა რიცხვს შეიტანს მასივში;
- ☐ b. [0,31]დიაპაზონიდან ალბუელ შემთხვევით რიცხვებს შეიტანს მასივში;
- ☐ c. 0-დან 30-მდეყველა რიცხვს შეიტანს მასივში;
- ☐ d. [0, 30] დიაპაზონიდან ალბუელ შემთხვევით რიცხვებს შეიტანს მასივში;

კითხვა 14

რომელი ფრაგმენტი ქმნის [0, 50] დიაპაზონიდან N შემთხვევითი რიცხვების მასივს?

აირჩიეთ ერთი:

- ☐ a. for(i=0; i<N; i++) a[i]=rand()%50
- ☐ b. for(i=0; i<N; i++) a[i]=rand()%(0-50)
- ☐ c. for(i=0; i<N; i++) a[i]=rand()%51

კითხვა 15

რა დაიბეჭდება ეკრანზე შემდეგი ფრაგმენტის შესრულების შემდეგ?

```
int main()
{
deque <int> dq;
cout << dq.empty() << endl;
dq.push_front(3);
dq.push_front(2);
```

```

dq.push_back(4);
dq.push_front(1);
dq.push_back(5);
cout << boolalpha << dq.empty() << endl;
cout << dq.size() << endl;
cout << dq.front() << endl;
cout << dq.back() << endl;
}

```

აირჩიეთ ერთი:



a.
false
true
5
1
5



b.
1
false
5
1
5



c.
0
false
5
1
5



d.
true
false
5
1
5

კითხვა 16

მოცემულია პროგრამის ფრაგმენტი:

```

{
dq.push_front(22);
dq.push_back(55);
dq.push_back(66);
}

```

რამდენი ელემენტია მოცემულ დეკში?

Answer:

კითხვა 17

მოცემულია პროგრამის ფრაგმენტი:

```
{  
dq.push_front(23);  
dq.push_back(55);  
dq.push_back(66);  
cout<<dq.front()<<endl;  
}
```

რა დაიბეჭდება მოცემული ფრაგმენტის შესრულების შემდეგ?

Answer:

კითხვა 18

რა დაიბეჭდება ეკრანზე შემდეგი ფრაგმენტის შესრულების შემდეგ?

```
int main()  
{  
stack <int> st;  
cout << boolalpha << st.empty() << endl;  
st.push(5);  
st.push(7);  
st.push(9);  
cout<<st.size()<<endl;  
cout<<st.top()<<endl;  
st.pop();  
cout<<st.size()<<endl;  
st.pop();  
cout<<st.size()<<endl;  
cout << boolalpha << st.empty() << endl;  
st.push(11);  
st.push(13);  
cout<<st.size()<<endl;  
return 0;  
}
```

აირჩიეთ ერთი:



- a.
true
3
5
2
1
false
3



- b.
false
3
5
2
1

true

3



c.

true

3

9

2

1

false

3



d.

false

3

9

2

1

true

3

კითხვა 19

მოცემულია პროგრამის ფრაგმენტი:

```
{
```

```
.....
```

```
st.push(22);
```

```
st.push(2);
```

```
st.push (19);
```

```
.....
```

```
}
```

რამდენი ელემენტია მოცემულ სტეკში?

Answer:

კითხვა 20

მოცემულია პროგრამის ფრაგმენტი:

```
{
```

```
.....
```

```
st.push(14);
```

```
st.push(2);
```

```
st.push (19);
```

```
.....
```

```
}
```

რომელია პირველი ელემენტი სტეკში?

Answer:

კითხვა 21

რა დაიბეჭდება ეკრანზე შემდეგი ფრაგმენტის შესრულების შემდეგ?

```
{
```

```
queue <int> q;
```



```

cout << boolalpha << q.empty() << endl;
q.push(17);
q.push(25);
q.push(9);
cout<<q.size()<<endl;
q.pop();
cout << boolalpha << q.empty() << endl;
cout<<q.front()<<endl;
cout<<q.back()<<endl;
q.push(11);
cout<<q.size()<<endl;
}

```

აირჩიეთ ერთი:



a.
true
3
false
9
25
3



b.
false
3
true
9
25
3



c.
false
3
true
25
9
3



d.
true
3
false
25
9
3

კითხვა 22

მოცემულია პროგრამის ფრაგმენტი:

```

{
queue <int> q;
q.push(95);
q.push(22);

```

```
q.push(27);
q.push(45);
q.push(52);
q.pop();
cout<<q.size()<<endl;
}
```

რა დაიბეჭდება მოცემული ფრაგმენტის შესრულების შემდეგ?

Answer:

კითხვა 23

მოცემულია პროგრამის ფრაგმენტი:

```
{
queue <int> q;
q.push(95);
q.push(22);
q.push(27);
q.push(45);
q.push(52);
q.pop();
cout<<q.front()<<endl;
}
```

რა დაიბეჭდება მოცემული ფრაგმენტის შესრულების შემდეგ?

Answer:

კითხვა 24

რა დაიბეჭდება მოცემული ფრაგმენტის შესრულების შედეგად?

```
int n =179;
int *p;
p = &n;
cout<<*p;
```

Answer:

კითხვა 25

გვაქვს `int b[3] = { 2, 4, 6 }`, `*bPtr=b`; რომელი ოპერატორი მოგვცემს შეცდომას?

აირჩიეთ ერთი:

- ☐ a. `bPtr = b+1;`
- ☐ b. `bPtr = &b[0];`
- ☐ c. `b++;`
- ☐ d. `bPtr++;`

კითხვა 26

რა შედეგს გამოიტანს შემდეგი პროგრამა:

```
void swap (int *, int *);
```

```
int main(){
    int a=15, b=25;
    swap(&a, &b);
    cout<<a<<b;

}
void swap (int *x, int *y){
    int dam;
    dam = *x;
    *x = *y;
    *y = dam;
}
```

აირჩიეთ ერთი:

- ☐ a. -15 -25
- ☐ b. 25 15
- ☐ c. 15 25
- ☐ d. პროგრამა არ იმუშავებს;

კითხვა 27

მოცემულია ცვლადების მისამართები: &a=63384,&b=64390,&c=64400. რა შედეგს გამოიტანს პროგრამა 32-ბიტის კომპიუტერებისთვის (int - იკავებს 4 ბაიტს, float იკავებს - 4 ბაიტს, double - 8 ბაიტს) :

```
int main()
{
    float a,*p1;
    int b,*p2;
    double c,*p3;
    a=2.5; b=3; c=2.96;
    p1=&a; p2=&b; p3=&c;
    p1++; p2++; p3++;
    cout<<"p1="<<p1<<" p2="<<p2<<" p3="<<p3;
}
```

აირჩიეთ ერთი:

- ☐ a. p1=3.5 p2=4 p3=B
- ☐ b. p1=63388 p2=64394 p3=64404
- ☐ c. p1=63388 p2=64394 p3=64408
- ☐ d. არცერთი პასუხი არ არის სწორი

კითხვა 28

რა დაიბეჭდება პროგრამის შესრულების შემდეგ:

```
#include <iostream>
using namespace std;
int main()
```

```
{
    int *p, x=5;
    p=&x;
    cout<<+>(*p);
    system("pause");
    return 0;
}
```

აირჩიეთ ერთი:

- ☐ a. x ცვლადის მისამართი
- ☐ b. 6
- ☐ c. 5
- ☐ d. კომპილაციის შეცდომაა
- ☐ e. 7

კითხვა 29

მიმთითებლის რომელი აღწერაა სწორია?

აირჩიეთ ერთი:

- ☐ a. int x;
- ☐ b. int *x;
- ☐ c. int &x;
- ☐ d. ptr x;

კითხვა 30

რას ითვლის შემდეგი პროგრამა?:

```
#include <iostream>
using namespace std;
unsigned int f(unsigned int n);
int main()
{
    unsigned int n = 5;
    unsigned int y = f(n);
    cout << n<<"! =" << y << endl;
    return 0;
}
unsigned int f(unsigned int n)
{
    unsigned int t;
    if (n <= 1)
        t = 1;
    else
        t = n * f(n - 1);
    return t;
}
```

აირჩიეთ ერთი:

- ☐ a. ბეჭდავს n-ის შებრუნებულ მნიშვნელობას;
- ☐ b. n-ის ფაქტორიალს;
- ☐ c. n-ის ციფრთა ჯამს;

კითხვა 31

მოცემულია სტრუქტურა, როგორ გამოვაცხადებთ ამ სტრუქტურის ტიპის x ცვლადს?

```
struct rectangle{  
    float length;  
    float width;  
};
```

აირჩიეთ ერთი:

- ☐ a. struct rectangle x;
- ☐ b. struct rectangle;
- ☐ c. struct x;
- ☐ d. struct x rectangle;

კითხვა 32

სტრუქტურა არის მონაცემთა ტიპი, რომელშიც

აირჩიეთ ერთი:

- ☐ a. ყველა ელემენტი უნდა იყოს მხოლოდ ტექსტური ტიპის;
- ☐ b. ყველა ელემენტი უნდა იყოს ერთნაირი ტიპის;
- ☐ c. ელემენტები უნდა იყონ მხოლოდ რიცხვითი ტიპის;
- ☐ d. ელემენტები შეიძლება იყოს სხვადასხვა ტიპის;

კითხვა 33

ფუნქცია ითვლის ნამდვილი რიცხვითი ტიპის მასივის იმ ელემენტების ჯამს, რომელთა ინდექსები იცვლება p-დან q-მდე (იგულისხმება, რომ $p < m$ და $q < m$). რომელი ბრძანებით შეიძლება მოცემული ფუნქციის გამოძახება?

```
const int m = 5;  
float sum(float [],int , int );  
int main()  
{  
    float mass[m]; int i,p,q;  
    float rez;  
    for (i=0;i<m;i++) cin>>mass[i];  
    cout<<"p?q?";  
    cin>>p>>q;  
    .....  
}
```

```
float sum(float a[],int p1,int q1){
    int k; float s =0;
    for(k=p1; k<=q1; k++)
        s +=a[k];
    return s;
}
```

აირჩიეთ ერთი:

- ☐ a. rez = sum(mass,p,q);
- ☐ b. rez = sum(mass[],p,q);
- ☐ c. rez = sum(a[],p,q);
- ☐ d. rez = sum(a,p,q);

კითხვა 34

რას აბრუნებს შემდეგი ფუნქცია:

```
double S(double a[], int N)
{
    double sum=0;
    for (int i=1; i<N; i+=2)
        sum+=a[i];
    return sum;
}
```

აირჩიეთ ერთი:

- ☐ a. კენტინდექსებიანი წევრების ჯამს;
- ☐ b. კენტინდექსებიანი დადებით წევრების ჯამს;
- ☐ c. ლუწინდექსებიანი წევრების ჯამს;
- ☐ d. ლუწინდექსებიანი დადებით წევრების ჯამს;
- ☐ e. ყველა რიცხვის ჯამს

კითხვა 35

რა მიიღება შემდეგი ფრაგმენტის შესრულების შედეგად?

```
const int N=5;
void printArray(int a[N]);
int main(){
    int mass[N]={10,-1,5,7,2};
    printArray(mass);
    . . .
}
void printArray(int a[]){
    inti;
    for(i=0; i<N; i++)
        cout<< a[i]<<" ";
}
```

პირივეთ ვრთი:



a. 10 -1 5 7 2



b. 10 5 2



c. 2 7 5 -1 10