

მონაცემთა სტრუქტურები და ალგორითმები (I-II კურსი)

I-II სემესტრი. 2020-2021 წელი

(შუასემესტრული გამოცდის ნიმუში)

კითხვა 1

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.
true
false
5
1
5



b.
0
false
5
1
5



c.
false
true
5
1
5



d.
1
false
5
1
5

კითხვა 2

მოცემულია პროგრამის ფრაგმენტი:

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

რამდენი ელემენტია მოცემულ ღვეშში?

Answer:

კითხვა 3

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 4

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
9
2
1
true
3



c.
false
3
5
2
1
true
3



d.
true
3
9
2
1

false

3

კითხვა 5

მოცემულია პროგრამის ფრაგმენტი:

```
{  
.....  
st.push(22);  
st.push(2);  
st.push (19);  
.....  
}
```

რამდენი ელემენტი მოცემულ სტეკში?

Answer:

კითხვა 6

მოცემულია პროგრამის ფრაგმენტი:

```
{  
.....  
st.push(14);  
st.push(2);  
st.push (19);  
.....  
}
```

რომელია პირველი ელემენტი სტეკში?

Answer:

კითხვა 7

რა დაბეჭდება 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
25
9
3



b.
true
3
false
9
25
3



c.
false
3

true
9
25
3
☐ d.
false
3
true
25
9
3

კითხვა 8

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 9

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 10

რა დაიბეჭდება მოცემული ფრაგმენტის შესრულების შედეგად?

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

Answer:

კითხვა 11

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

აირჩიეთ ერთი:

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



d. b++;

კითხვა 12

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. პროგრამა არ იმუშავებს;



b. 15 25



c. -15 -25



d. 25 15

კითხვა 13

მოცემულია ცვლადების მისამართები: &a=63384,&b=64390,&c=64400. რა შედეგს გამოიტანს პროგრამა 32-ბიტიანი 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. არცერთი პასუხი არ არის სწორი



c. p1=63388 p2=64394 p3=64408



d. p1=63388 p2=64394 p3=64404

კითხვა 14

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

კითხვა 15

მიმთითებლის რომელი აღწერაა სწორია?

აირჩიეთ ერთი:

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

კითხვა 16

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

კითხვა 17

მოცემულია სტრუქტურა, როგორ გამოვაცხადებთ ამ სტრუქტურის ტიპის x ცვლადს?

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

აირჩიეთ ერთი:

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



c. struct rectangle x;



d. struct x;

კითხვა 18

სტრუქტურა არის მონაცემთა ტიპი, რომელშიც

აირჩიეთ ერთი:



a. ელემენტები შეიძლება იყოს სხვადასხვა ტიპის;



b. ყველა ელემენტი უნდა იყოს ერთნაირი ტიპის;



c. ელემენტები უნდა იყონ მხოლოდ რიცხვითი ტიპის;



d. ყველა ელემენტი უნდა იყოს მხოლოდ ტექსტური ტიპის;

კითხვა 19

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(a,p,q);



c. rez = sum(mass[],p,q);



d. rez = sum(a[],p,q);

კითხვა 20

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

კითხვა 21

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]<<" ";
}
```

აირჩიეთ ერთი:

- ☐ a. 2 7 5 -1 10
- ☐ b. 10 5 2
- ☐ c. 10 -1 5 7 2