

## კითხვა 1

ჯერ პასუხგაცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

დარჩენილი დრო 0:44:57

დამალე

მოცემულია პროგრამის ფრაგმენტი:

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

რამდენი ელემენტია მოცემულ დეკში?

Answer:



## კითხვა 2

ჯერ პასუხგაუცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

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



### კითხვა 3

ჯერ პასუხგაუცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

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;
}
```

### პირჩიეთ 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 4

ჯერ პასუხგაუცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

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

### აირჩიეთ ერთი:

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



## კითხვა 5

ჯერ პასუხგაუცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

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



## კითხვა 6

ჯერ პასუხგაუცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

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 7

ჯერ პასუხაუცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

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 8

ჯერ პასუხაუცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

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 9

ჯერ პასუხგაცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

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]<<" ";
}
```

**პირჩინეთ პასუხი:**

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



## კითხვა 10

ჯერ პასუხგაცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

მიმთითებლის რომელი აღწერაა სწორია?

### აირჩიეთ ერთი:

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



## კითხვა 11

ჯერ პასუხგაცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

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;
}
```

**პირველი პრთი:**

☐ a.

true  
3  
9  
2  
1  
false  
3



☐ b.

true  
3  
5  
2

1  
false  
3

☐ c.

false  
3  
9  
2  
1  
true  
3

☐ d.

false  
3  
5  
2  
1  
true  
3

## კითხვა 12

ჯერ პასუხგაუცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

მოცემულია პროგრამის ფრაგმენტი:

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

რამდენი ელემენტია მოცემულ სტეკში?

Answer:

### კითხვა 13

ჯერ პასუხგაუცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

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 14

ჯერ პასუხგაუცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

მოცემულია პროგრამის ფრაგმენტი:

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

რომელია პირველი ელემენტი სტეკში?

Answer:



## კითხვა 15

ჯერ პასუხაუცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

მოცემულია სტრუქტურა, როგორ გამოვაცხადებთ ამ სტრუქტურის ტიპის x ცვლადს?

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

**აირჩიეთ ერთი:**

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



## კითხვა 16

ჯერ პასუხგაუცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

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 17

ჯერ პასუხგაცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

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



## კითხვა 18

ჯერ პასუხგაცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

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;
}
```

**პირჩიეთ ერთი:**

☐ a.

1  
false  
5  
1  
5

☐ b.

0  
false  
5  
1  
5

☐ c.

true  
false  
5



1

5

☐ d.

false

true

5

1

5

## კითხვა 19

ჯერ პასუხაუცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

სტრუქტურა არის მონაცემთა ტიპი, რომელშიც

### აირჩიეთ ერთი:

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



## კითხვა 20

ჯერ პასუხგაცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

რა დაიბეჭდება მოცემული ფრაგმენტის შესრულების შედეგად?

```
int n =179;
```

```
int *p;
```

```
p = &n;
```

```
cout<<*p;
```

Answer:



## კითხვა 21

ჯერ პასუხგაცემელი

გამოანგარიშებულია შემდეგი რაოდენობიდან 1.00

v1 (latest)

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;  
}
```

**პირჩინეთ ერთი:**

☐ a.

true  
3  
false  
25  
9  
3

☐ b.

false  
3  
true  
25  
9  
3

☐ c.

true  
3



false

9

25

3

☐ d.

false

3

true

9

25

3

