

**Question 1**

Time left 0:44:40

Hide

Not yet answered

Marked out of 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;
}
```

**Select one:**

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



**Question 2**

Not yet answered

Marked out of 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: 

**Question 3**

Not yet answered

Marked out of 1.00

v1 (latest)

მოცემულია პროგრამის ფრაგმენტი:

{

.....

st.push(22);

st.push(2);

st.push (19);

.....

}

რამდენი ელემენტია მოცემულ სტეკში?

Answer:

**Question 4**

Not yet answered

Marked out of 1.00

v1 (latest)

მოცემულია პროგრამის ფრაგმენტი:

{

dq.push\_front(22);

dq.push\_back(55);

dq.push\_back(66);

}

რამდენი ელემენტია მოცემულ დეკში?

Answer:



**Question 5**

Not yet answered

Marked out of 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;
}
```

**Select one:** 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



**Question 6**

Not yet answered

Marked out of 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;
}
```

**Select one:**

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



**Question 7**

Not yet answered

Marked out of 1.00

v1 (latest)

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

**Select one:**

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



**Question 8**

Not yet answered

Marked out of 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;
}
```

**Select one:**

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



**Question 9**

Not yet answered

Marked out of 1.00

v1 (latest)

ორობითი ძეგლის დროს საჭიროა გვექონდეს

**Select one:**

- a. ზრდადობის ან კლებადობის მიხედვით დახარისხებული მასივი
- b. დაუხარისხებელი მასივი
- c. არცერთი პასუხი არ არის სწორი

**Question 10**

Not yet answered

Marked out of 1.00

v1 (latest)

სტრუქტურა არის მონაცემთა ტიპი, რომელშიც

**Select one:**

- 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;



**Question 11**

Not yet answered

Marked out of 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;
}
```

**Select one:**

- 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



**Question 12**

Not yet answered

Marked out of 1.00

v1 (latest)

მოცემულია სტრუქტურა, როგორ გამოვაცხადებთ ამ სტრუქტურის ტიპის x ცვლადს?

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

**Select one:**

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



**Question 13**

Not yet answered

Marked out of 1.00

v1 (latest)

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 შეიტანთ 37-ს

**Select one:**

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

**Question 14**

Not yet answered

Marked out of 1.00

v1 (latest)

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;
        }
```

**Select one:**

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

**Question 15**

Not yet answered

Marked out of 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;
}
```

**Select one:** a.

1  
false  
5  
1  
5

 b.

0  
false  
5  
1  
5

 c.

true  
false  
5



1

5

d.

false

true

5

1

5

### Question 16

Not yet answered

Marked out of 1.00

v1 (latest)

მოცემულია პროგრამის ფრაგმენტი:

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

რომელია პირველი ელემენტი სტეკში?

Answer:



**Question 17**

Not yet answered

Marked out of 1.00

v1 (latest)

მიმთითებლის რომელი აღწერაა სწორია?

**Select one:**

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

**Question 18**

Not yet answered

Marked out of 1.00

v1 (latest)

რა დაიბეჭდება მოცემული ფრაგმენტის შესრულების შედეგად?

```
int n =179;
```

```
int *p;
```

```
p = &n;
```

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

Answer: 

**Question 19**

Not yet answered

Marked out of 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: 

**Question 20**

Not yet answered

Marked out of 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;
```

```
}
```

**Select one:**

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



**Question 21**

Not yet answered

Marked out of 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]<<" ";
}
```

**Select one:**

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



**Question 22**

Not yet answered

Marked out of 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: 

**Question 23**

Not yet answered

Marked out of 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;
```

```
}
```

**Select one:**

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



**Question 24**

Not yet answered

Marked out of 1.00

v1 (latest)

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;
}
```

**Select one:**

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

