

## მონაცემთა სტრუქტურები და დაპროგრამების საფუძვლები 2

### II სემესტრი 2017-2018 წელი

#### (I შუალედური ტესტირების ნიმუში)

##### კითხვა 1

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. მარტივი გადანაცვლების ალგორითმი
- b. ბუშტისებრი-ჩადირვის ალგორითმი
- c. კომბინირებული ალგორითმი
- d. არც ერთი

##### კითხვა 2

რას აკეთებს პროგრამის შემდეგი ფრაგმანტი?

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

### კითხვა 3

ორობითი ძებნის დროს საჭიროა გვექონდეს

აირჩიეთ ერთი:

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

### 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 შეიტანთ 37-ს

აირჩიეთ ერთი:

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

#### კითხვა 5

როგორი ტიპისაა ფუნქცია, თუ მას დასაბრუნებელი მნიშვნელობის ტიპი არა აქვს მითითებული?

აირჩიეთ ერთი:

- a. void
- b. int
- c. სინტაქსური შეცდომაა
- d. double

#### კითხვა 6

რომელია ბლოკის შიგნით აღწერილი ცვლადის მოქმედების არე:

Select one or more:

- a. ამ აღწერის შემცველი ბლოკი
- b. ეს ბლოკი მასში ჩადგმული ქვებლოკების გარეშე
- c. ცვლადის აღწერიდან ბლოკის დამამთავრებელ მარჯვენა ფრჩხილამდე
- d. ყველა ის ფუნქცია, რომელშიც ეს ცვლადია გამოყენებული

#### კითხვა 7

რომელი თავი (header) ფაილია საჭირო ნაკადის პარამეტრიზებული მანიპულატორის setprecision() გამოყენებისთვის?

აირჩიეთ ერთი:

- a. iostream
- b. iomanip
- c. conio
- d. cmath

#### კითხვა 8

რა ტიპის მონაცემს აბრუნებს ქვემოთ მოცემული პროტოტიპის შესაბამისი ფუნქცია?

```
int func(char x, float v, double t);
```

აირჩიეთ ერთი:

- a. double
- b. char
- c. int
- d. float

**კითხვა 9**

რა შედეგი დაიბეჭდება ეკრანზე პროგრამის შემდეგი ფრაგმენტის შესრულების შემდეგ?

```
int my_abs(int);
int main() {
int y=5;
cout<< my_abs(y);
getch();
return 0;
}
int my_abs(int x){
if(x >= 0) return x;
return -x;
}
```

აირჩიეთ ერთი:

- a. 5
- b. -5
- c. სინტაქსური შეცდომა
- d. true

**კითხვა 10**

მოცემულია ფუნქცია my\_pow რომელსაც აქვს სახე:

```
float my_pow(float x, int n) {
float d = 1; int i;
if(n == 0) return 1;
else if(n > 0) {
for(i = 1; i <= n; i++)
d *= x;
return d;
}
else
for(i = -1; i >= n; i--) {
d *= 1/x;
return d;
}
}
```

რა მნიშვნელობას მიიღებს p ცვლადი თუ მისი გამოძახება პროგრამაში ხდება შემდეგნაირად:

```
float p= my_pow(3,2);
```

Answer:

### კითხვა 11

რას გამოიტანს შემდეგი პროგრამა:

```
double m(double x, double y);
void main()
{
    double a=7, b=3.5, c=2.5, d=9;
    cout << m(m(a,b), m(c,d))<<endl;
    system ("pause");
}
double m(double x, double y)
{
    return y>x ? y: x;
}
```

აირჩიეთ ერთი:

- a. 9
- b. 2.5
- c. 3.5
- d. 2
- e. რიცხვების ჯამს

### კითხვა 12

რა დაიბეჭდება შემდეგი ფრაგმენტის შესრულების შედეგად:

```
int main(){
    f1();
}
void f1(void){
    int a;
    for(a=1; a<30; a+=2)
        cout<<a<<endl;
}
```

აირჩიეთ ერთი:

- a. ლუწი რიცხვები 1-დან 30-მდე;
- b. კენტი რიცხვები 1-დან 30-მდე;
- c. 3-ის ჯერადი რიცხვები 1-დან 30-მდე;
- d. ყველა რიცხვი 1-დან 30-მდე;

### კითხვა 13

რა მნიშვნელობას მიიღებს n შემდეგი ფრაგმენტის შესრულების შედეგად:

```
int main(){
    int k=7, n;
    n =square(k);
}
```

```
}  
int square(int a){  
    return a*a;  
}
```

სიჩქარე ვართ:

- a. 7
- b. 35
- c. 49
- d. 17