

|               |  |
|---------------|--|
| Indeks        |  |
| Prezime i ime |  |

1. Šta ispisuje sljedeći kôd?

```
#include <iostream>
using namespace std;

class KD {
    public:    KD(int);
             ~KD();
    private: int data;
};

KD::KD(int x) { data=x; cout<<data; }

KD::~KD() { cout<<data; }

void create( void )
{ KD kd5(1); static KD kd6(2); }

KD kd1(3);

int main()
{
    KD kd2(4); create(); KD kd4(5);
    return 0;
}
```

Rješenje:

2. Objasniti i na konkretnom primjeru ilustrovati inicijalizaciju konstantnih atributa objekta.

3. Šta ispisuje sljedeći kôd?

```
#include <iostream>
using namespace std;

int dat=1;
class A
{
    int pod;
    public:
    A(): pod(dat++) { pod++; }
    ~A() { dat--; cout << "*"; }
    A& operator++()
        { cout<<"a"<<pod--; return *this; }
    A operator++(int)
        { cout<<"b"<<pod++; return *this; }
    A& operator--()
        { cout<<"c"<<pod++; return *this; }
    A operator--(int)
        { cout<<"d"<<pod--; return *this; }
};

int main()
{
    A y; ++--y++; cout << endl;
    return 0;
}
```

Rješenje:

4. Objasniti i na konkretnom primjeru ilustrovati preklapanje operatora <<.

5. Šta ispisuje sljedeći kôd?

```
#include <iostream>
using namespace std;

class X
{
public:
    X(int i=1) : x(i) { cout<<"X"; }
protected: int x;
};

class A : virtual public X
{
public:
    A(int i=10) : X(i) { cout<<"A"; }
    void print() { cout<<x<<endl; }
};

class B : virtual public X
{
public:
    B(int i=20) : X(i) { cout<<"B"; }
};

class C : public A, public B
{
public:
    C(int i=50) : X(i) { cout<<"C"; }
};

int main()
{
    A a; a.print();
    C c; c.print();
    return 0;
}
```

Rješenje:

6. Objasniti pojam virtuelnog destruktora i ilustrovati njegovo korištenje.

7. Šta ispisuje sljedeći kôd?

```
#include <iostream>
using namespace std;
class A {};
class B : public A {};
class C : public B {};
int main()
{
    A a; B b; C c;
    for (int i=1; i<3; i++) {
        try {
            for (int j=i+1; j<5; j++) {
                try {
                    switch (j%3) {
                        case 0: throw c;
                        case 1: throw b;
                        default: throw a;
                    }
                }
                catch(C) { cout<<"C"; }
                catch(B) { cout<<"B"; throw a; }
                catch(A) { cout<<"A"; throw b; }
            }
        }
        catch(A) { cout<<"A"<<endl; }
        catch(B) { cout<<"B"<<endl; }
        catch(C) { cout<<"C"<<endl; }
    }
    return 0;
}
```

Rješenje:

8. Definirati generičku funkciju za ispisivanje niza podataka sa  $n$  elemenata, pri čemu je parametar šablona tip podataka u nizu. U glavnoj funkciji ilustrovati ispis nekog niza pozivom odgovarajuće funkcije.