



REPUBLIKA E SHQIPËRISË
MINISTRIA E ARSIMIT
QENDRA E SHËRBIMEVE ARSIMORE

OLIMPIADA KOMBËTARE E INFORMATIKËS
NË ARSIMIN E MESËM TË LARTË

Faza e dytë

Klasa 11

17 janar 2026

Udhëzime për nxënësin:

- Olimpiada fillon në orën 10:00 dhe mbaron në orën 13:00.
- Testi përmban 5 pyetje.
- Për të zgjidhur secilin ushtrim nxënësi mund të përdorë gjuhën e programimit C/C++.

Për përdorim nga komisioni i vlerësimit

Pyetja	1	2	3	4	5
	8 pikë	12 pikë	10 pikë	8 pikë	12 pikë
Pikët e fituara					

Totali i pikëve të fituara

KOMISIONI I VLERËSIMIT

1.....

2.....

1. Shkruani një program për të shfaqur të gjitha kombinimet e mundshme të një vargu të dhënë hyrës. Nëse karakteret përsëriten më shumë se një herë, rezultatet nuk duhet të përsëriten. **8 pikë**

Shembull:

input = aba

output:aba

aab

baa

Zgjidhje:

```
#include <iostream>
#include <algorithm>
#include <string>
using namespace std;
int main() {
    string hyrja;
    cout << "Jepni vargun : ";
    cin >> hyrja;
    sort(hyrja.begin(), hyrja.end());
    cout << "Kombinimet unike janë:" << endl;
    do {
        cout << hyrja << endl;
    } while (next_permutation(hyrja.begin(), hyrja.end()));
    return 0;
}
```

2. Jepet një varg me n numra të plotë dhe një numër M pozitiv. Duhet të hiqen M elementë tek nga vargu (jo patjetër të njëpasnjëshëm). Shkruani një program që gjen shumën maksimale që mund të mbetet pas eliminimit të M elementëve. **(12 pikë)**

Zgjidhje:

```
#include <iostream>
```

```
#include <vector>
```

```
#include <algorithm>

#include <numeric>

using namespace std;

int main() {

    int n, M;

    cout << "Jepni numrin e elementeve (n) dhe numrin e elementeve per t'u hequr (M): ";

    cin >> n >> M;

    vector<int> numrat_tek;

    long long shuma_cift = 0;

    int vlera;

    cout << "Jepni elementet e vargut: ";

    for (int i = 0; i < n; i++) {

        cin >> vlera;

        if (vlera % 2 != 0) {

            // Nese numri eshte tek, e ruajme ne nje liste te vecante

            numrat_tek.push_back(vlera);

        } else {

            // Nese eshte cift, e shtojme direkt te shuma sepse nuk do te hiqet

            shuma_cift += vlera;

        }

    }

    // Kontrollojme nese ka mjaftueshem numra tek per t'u hequr

    if (numrat_tek.size() < M) {

        cout << "Gabim: Vargu nuk ka " << M << " elemente tek." << endl;

        return 1;

    }

    // Rendisim numrat tek nga me i vogli te me i madhi

    sort(numrat_tek.begin(), numrat_tek.end());

    // Heqim M elementet e pare (ata me vleren me te vogel)

    // Duke filluar mbledhjen nga indeksi M deri ne fund
```

```
long long shuma_tek_mbetur = 0;

for (int i = M; i < numrat_tek.size(); i++) {
    shuma_tek_mbetur += numrat_tek[i];
}

long long shuma_maksimale = shuma_cift + shuma_tek_mbetur;

cout << "Shuma maksimale e mbetur eshte: " << shuma_maksimale << endl;

return 0;
}
```

3. Shkruani një program që implementon një sistem rezervimi biletash ku përdoruesi mund të: **(10 pikë)**

- a) shtojë rezervime;
- b) anulojë rezervime;
- c) shfaqë vendet e lira;
- d) shfaqë vendet e zëna.

Zgjidhje:

```
#include <iostream>

#include <vector>

using namespace std;

// Funksion për të shfaqur menunë

void shfaqMenu() {
    cout << "\n--- SISTEMI I REZERVIMIT TE BILETAVE ---" << endl;
    cout << "1. Shto Rezervim" << endl;
    cout << "2. Anulo Rezervim" << endl;
    cout << "3. Shfaq Vendet e Lira" << endl;
    cout << "4. Shfaq Vendet e Zena" << endl;
    cout << "5. Mbylle Programin" << endl;
    cout << "Zgjidhni nje opsion: ";
}

int main() {
```

```
int nrVendeve = 10; // Mund ta ndryshoni sipas deshires

vector<bool> vendet(nrVendeve, false); // Fillimisht te gjitha jane false (te lira)

int zgjedhja, nrVendi;

do {
    shfaqMenu();

    cin >> zgjedhja;

    switch (zgjedhja) {
        case 1: // Shto rezervim
            cout << "Jepni numrin e vendit (0-" << nrVendeve - 1 << "): ";
            cin >> nrVendi;

            if (nrVendi >= 0 && nrVendi < nrVendeve) {
                if (!vendet[nrVendi]) {
                    vendet[nrVendi] = true;

                    cout << "Rezervimi u krye me sukses!" << endl;
                } else {
                    cout << "GABIM: Ky vend eshte i zene!" << endl;
                }
            } else {
                cout << "Vendi nuk ekziston!" << endl;
            }

            break;

        case 2: // Anulo rezervim
            cout << "Jepni numrin e vendit per te anuluar: ";
            cin >> nrVendi;

            if (nrVendi >= 0 && nrVendi < nrVendeve) {
                if (vendet[nrVendi]) {
                    vendet[nrVendi] = false;

                    cout << "Rezervimi u anulua me sukses." << endl;
                } else {
                    cout << "Ky vend eshte tashme i lire." << endl;
                }
            }
        }
    }
}
```

```
}  
  
} else {  
    cout << "Vendi nuk ekziston!" << endl;  
}  
  
break;  
  
case 3: // Shfaq vendet e lira  
    cout << "Vendet e lira: ";  
    for (int i = 0; i < nrVendeve; i++) {  
        if (!vendet[i]) cout << "[" << i << " ] ";  
    }  
    cout << endl;  
    break;  
  
case 4: // Shfaq vendet e zena  
    cout << "Vendet e zena: ";  
    bool kaTeZena = false;  
    for (int i = 0; i < nrVendeve; i++) {  
        if (vendet[i]) {  
            cout << "[" << i << " ] ";  
            kaTeZena = true;  
        }  
    }  
    if (!kaTeZena) cout << "Asnje vend nuk eshte i zene."  
    cout << endl;  
    break;  
}  
  
} while (zgjedhja != 5);  
  
return 0;  
}
```

4. Shkruani një program i cili regjistron studentët dhe numrin e projekteve që ata marrin pjesë. Çdo projekt paguhet 100\$. Programi afishon emrin e studentit me më shumë bonuse, numrin e projekteve dhe shumën totale që fiton. (8 pikë)

Zgjidhje:

```
#include <iostream>

#include <string>

using namespace std;

int main() {

    int n;

    cout << "Jep numrin e studenteve: ";

    cin >> n;

    string emri, maxEmri;

    int projekte, maxProjekte = 0;

    for (int i = 0; i < n; i++) {

        cout << "\nStudenti " << i + 1 << endl;

        cout << "Emri: ";

        cin >> emri;

        cout << "Numri i projekteve: ";

        cin >> projekte;

        if (projekte > maxProjekte) {

            maxProjekte = projekte;

            maxEmri = emri;

        }

    }

    int shuma = maxProjekte * 100;

    cout << "\n--- STUDENTI ME ME SHUME BONUS ---\n";

    cout << "Emri: " << maxEmri << endl;

    cout << "Numri i projekteve: " << maxProjekte << endl;

    cout << "Shuma totale e fituar: " << shuma << " $\n";

    return 0;
```

}

5. Jepet një matricë $n \times m$ me vlera 0 dhe 1, ku $1 \leq n \leq 100$ dhe $1 \leq m \leq 100$. Nga qeliza (1,1) mund të lëvizet vetëm djathtas dhe poshtë. Qelizat me vlerën 1 janë të bllokuara. Shkruani një program që gjen numrin e rrugëve për të arritur në (n,m). (12 pikë)

Zgjidhje:

```
#include <iostream>

#include <vector>

using namespace std;

int main() {
    int n, m;

    cout << "Jepni dimensionet n dhe m: ";

    cin >> n >> m;

    int grid[101][101];

    long long dp[101][101] = {0};

    cout << "Jepni matricen (0 per rruge, 1 per pengese):" << endl;

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            cin >> grid[i][j];
        }
    }

    // Nese pika e nisjes ose e mberritjes eshte e bllokuar
    if (grid[0][0] == 1 || grid[n-1][m-1] == 1) {
        cout << "Numri i rrugeve: 0" << endl;
        return 0;
    }

    // Rasti baze: pika e nisjes
    dp[0][0] = 1;

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            // Nese jemi te pika e nisjes, e kemi llogaritur me lart
```



```
if (i == 0 && j == 0) continue;

// Nese qeliza eshte pengese, rruget deri ketu jane 0

if (grid[i][j] == 1) {
    dp[i][j] = 0;
} else {
    long long lart = (i > 0) ? dp[i-1][j] : 0;
    long long majtas = (j > 0) ? dp[i][j-1] : 0;
    dp[i][j] = lart + majtas;
}
}
}

cout << "Numri total i rrugëve të mundshme: " << dp[n-1][m-1] << endl;

return 0;
}
```