

Oglinda lecției

Data: 10.03.2025

Clasa: 10 MI 2

Lecția: Interpretare test șir de caractere

Tipul lecției: Recapitulare și sintetizare

Activitatea din lecție	Observații
<p>Titlul lecției: Interpretare test șir de caractere</p> <p>Rezumat: Discutarea testului șiruri de caractere.</p> <p>Textul problemelor: Se citește din fișierul date.in, de pe prima linie, un număr natural n și de pe următoarele n rânduri n cuvinte, câte unul pe o linie. Cuvintele pot fi formate din litere mici, litere mari sau caractere cifre. Se cere să se scrie următoarele subprograme și apelurile corespunzătoare pentru justificarea funcționalității.</p> <ol style="list-style-type: none">1. Subprogram care citește șirul de cuvinte din fișier. Afișați șirul de cuvinte cu un spațiu între ele.2. Subprogram care determină, pentru un cuvânt dat ca parametru, dacă conține cel puțin un caracter cifră. Afișați cuvintele din șir care respectă proprietatea, de la primul cuvânt la ultimul.3. Subprogram care determină, pentru datele citite din fișier, dacă toate cuvintele conțin doar caractere de tip literă. Afișați un mesaj de tipul DA sau NU.4. Subprogram care afișează, pentru datele citite din fișier, un șir de caractere format din prima literă a fiecărui cuvânt.5. Subprogram care determină câte cuvinte au prima literă vocală. Afișați în apel valoarea determinată. <p>Ideea de rezolvare:</p> <ol style="list-style-type: none">1.	<p>Obiectivele lecției: O1: Recunoașterea funcțiilor și algoritmilor de bază pentru manipularea șirurilor de caractere. O2: Dezvoltarea atenției la detalii și la gestionarea corectă a indicilor în șiruri.</p> <p>Conținuturi:</p> <ul style="list-style-type: none">- grad de dificultate: mediu- importanță: esențial pentru descoperirea greșelilor efectuate la test- structurarea: inductivă <p>Evenimentele lecției: captarea atenției, fixarea cunoștințelor, evaluarea cunoștințelor</p> <p>Strategia didactică: rezolvarea de probleme, conversația, explicația</p> <p>Evaluare: observație curentă, evaluare orală</p>

```

void citireAfisareCuvinte(char cuvinte[n][n], int &n) {
    ifstream fin("date.in");
    fin >> n;
    for (int i = 0; i < n; i++) {
        fin >> cuvinte[i];
        cout << cuvinte[i] << " ";
    }
    cout << endl;
    fin.close();
}

```

2.

```

bool contineCifra(char cuvint[n]) {
    for (int i = 0; cuvint[i] != '\0'; i++) {
        if (cuvint[i] >= '0' && cuvint[i] <= '9') {
            return true;
        }
    }
    return false;
}

void afisareCuvinteCuCifre(char cuvinte[n][n], int n) {
    for (int i = 0; i < n; i++) {
        if (contineCifra(cuvinte[i])) {
            cout << cuvinte[i] << " ";
        }
    }
    cout << endl;
}

```

3.

```

bool toateDoarLitere(char cuvinte[n][n], int n) {
    for (int i = 0; i < n; i++) {
        for (int j = 0; cuvinte[i][j] != '\0'; j++) {
            if (!((cuvinte[i][j] >= 'a' && cuvinte[i][j] <= 'z')
                || (cuvinte[i][j] >= 'A' && cuvinte[i][j] <= 'Z')))) {
                return false;
            }
        }
    }
    return true;
}

```

4.

```

void afisarePrimaLitera(char cuvinte[n][n], int n) {
    for (int i = 0; i < n; i++) {
        cout << cuvinte[i][0];
    }
    cout << endl;
}

```

5.

```
bool esteVocala(char c) {  
    return c == 'a' || c == 'e'  
        || c == 'i' || c == 'o' || c == 'u' ||  
        c == 'A' || c == 'E'  
        || c == 'I' || c == 'O' || c == 'U';  
}  
  
int numarCuvinteVocale(char cuvinte[n][n], int n) {  
    int contor = 0;  
    for (int i = 0; i < n; i++) {  
        if (esteVocala(cuvinte[i][0])) {  
            contor++;  
        }  
    }  
    return contor;  
}
```