

#### Problema 4:

Scrieți codul necesar pentru ca următoarea funcție **main()**

```
int main() {
    // Propozitie p; // acest cod NU ar trebui sa compileze
    Propozitie p1("Azi am cumparat morcovi de 10 ron");
    cout << "Cuvinte in propozitie: " << p1["count"] << endl;
    cout << "Numarul total de caractere in propozitie: " << p1["total_chars"] << endl;
    cout << "Numarul total de vocale in propozitie: " << p1["vowals"] << endl;
    cout << "Numarul total de numere in propozitie: " << p1["numbers"] << endl;
    cout << "Primul cuvant: " << p1[0] << endl;
    cout << "Ultimul cuvant: " << p1[-1] << endl;
    cout << "Penultimul cuvant: " << p1[-2] << endl;
    p1.Set("Am cumparat 10 morcovi, 12 castraveti si 14 rosii.");
    cout << "Numarul total de numere in propozitie: " << p1["numbers"] << endl;
    cout << "Numarul total de caractere in propozitie: " << p1["total_chars"] << endl;
    cout << "Al patrulea cuvant: " << p1[3] << endl;
    cout << "Numarul de cuvinte formate doar din doua caractere : " << p1(2) << endl;
    return 0;
}
```

să afișeze pe ecran următoarele:

Cuvinte in propozitie: 7  
Numarul total de caractere in propozitie: 27  
Numarul total de vocale in propozitie: 11  
Numarul total de numere in propozitie: 1  
Primul cuvant: Azi  
Ultimul cuvant: ron  
Penultimul cuvant: 10  
Numarul total de numere in propozitie: 3  
Numarul total de caractere in propozitie: 41  
Al patrulea cuvant: morcovi  
Numarul de cuvinte formate doar din doua caractere : 5

Azi,am,cumparat,morcovi,de,10,ron  
27 = suma lungimilor tuturor cuvintelor  
Azi am cumparat morcovi de 10 ron  
Cuvantul 10

Cuvintele 10, 12 si 14

Cuvintele: Am, 10, 12, si, 14

#### Observatii:

- Puteți folosi obiecte din **STL** pentru a rezolva aceasta problema. **NU aveți** voie să utilizați funcții specifice din `string.h` (`strcpy`, `strtok`, etc). Codul care conține astfel de funcții nu se punctează.
- Utilizați funcția `Set` în constructor
- Operatorul de indexare cu parametru de tipul *int* accepta și valori negative, astfel: **-1** ⇒ ultimul cuvânt, **-2** ⇒ penultimul cuvânt, s.a.m.d. Dacă se da un index în afara celor acceptați se returnează `nullptr`.
- Operatorul de indexare cu parametrul ***numbers*** returnează câte cuvinte sunt formate doar din cifre
- Operatorul de indexare cu parametrul ***vowals*** returnează numărul total de vocale din propoziție.
- Asigurați-vă ca specificațiile sunt respectate (ex: codul care nu ar trebui să compileze, NU compilează).

#### Barem:

- **[5p]** Fiecare clasă este scrisă folosind un fișier heder și unul `cpp`. Codul din *main* este într-un fișier separat.
- **[2p]** Constructorii de la clasa *Propozitie*
- **[10p]** Funcția `Set` din clasa *Propozitie*
- **[8p]** operatorul de indexare cu parametru de tipul *const char \**. Cate 1 pct pentru parametri *count* și *total\_chars*, respectiv 3 pct pentru parametri *vowals* și *numbers*
- **[4p]** operatorul de apel de funcție
- **[6p]** operatorul de indexare cu parametru de tip *int*