

Universiteti i Prishtinës
Fakulteti i Inxhinierisë Elektrike dhe Kompjuterike



Algoritmet dhe struktura e të dhënave

Vehbi Neziri

FIEK, Prishtinë 2013

Java e 3-të

- Funkcionet e mbingarkuara (overloaded)
 - Deklarimi
 - Thirrja
- Sinonimet
 - Përcaktimi i sinonimeve
 - Sinonimet lokale dhe globale
- Numërimet (enumerations)
 - Përcaktimi i grupit
 - Shfrytëzimi i grupit

Funksionet e mbingarkuara

- ❑ **Funksione të mbingarkuara** – dy ose më shumë funksione me të njëjtin emër, por me parametra të ndryshëm.
 - ❑ Numri i ndryshëm i parametrave
 - ❑ Tipi i ndryshëm i të dhënave

```
int main()
{
    printo(2);
    printo(2,5);
}
```

```
void printo(int x)
{
    cout << "Integer: " << x <<
endl;
}
```

```
void printo(float x, int y)
{
    cout << "Integer: " << x << "
dhe " << y << endl;
}
```

Shembull 3.1

- Të shkruhet programi i cili llogarit sipërfaqen e rrethit dhe drejtkëndëshit përmes funksionit të mbingarkuar `siperfaqja()`.

```
1 #include<iostream>
2 using namespace std;
3
4 #define pi 3.14159265359
5
6 void siperfaqja(int r)
7 {
8     cout<<"Sipërfaqja e rrethit:"<<pi*r*r;
9 }
10
11 void siperfaqja(int a,int b)
12 {
13     cout<<"Sipërfaqja e drejtkëndëshit:"<<a*b;
14 }
15
16 int main()
17 {
18     int a,b,r;
19
20     cout<<"Rrezja r= ";
21     cin>>r;
22     siperfaqja(r);
23
24     cout<<"\nKrahët a=?b=?";
25     cin>>a>>b;
26     siperfaqja(a,b);
27     cout<<endl;
28     system("pause");
29     return 0;
30 }
```

Shembull 3.2

- Të shkruhet programi për llogaritjen e shprehjes së mëposhtme përmes funksionit të mbingarkuar `shuma()`.

$$y1 = \sum_{i=0}^n (4i + 2)$$

$$y2 = (x + 3)^2 + \sum_{i=3}^{2n} (3i - 2x)$$

```
1  #include <iostream>
2  using namespace std;
3
4  double shuma(int n)
5  {
6      double s=0;
7      int i;
8      for (i=0;i<=n;i++)
9          s=s+(4*i+2);
10     return s;
11 }
12
13 double shuma(int n,double x)
14 {
15     double s=0;
16     int i;
17     for (i=3;i<=2*n;i++)
18         s=s+(3*i-2*x);
19     return s;
20 }
21
22 int main()
23 {
24     double x,y1,y2;
25     cout << "Vlera x=";
26     cin >> x;
27     cout << "\nVlerat e llogaritura\n\n";
28     y1=shuma(10);
29     cout << " y=" << y1;
30
31     y2=pow(x+3,2)*shuma(15,x);
32     cout << "\n z="<< y2 << "\n";
33     return 0;
34 }
```

Sinonimet

- ❑ Për tipet e të dhënave mund të krijohen sinonime ose pseudonime duke përdorur deklaratën `typedef`.



- ❑ `typedef` nuk krijon tip të ri të dhënave.
- ❑ `typedef` krijon një sinonim për tipin ekzistues.
- ❑ Sinonimet mund të jenë lokale dhe globale.

Shembull 3.3

- Të shkruhet programi për llogaritjen e faktorielit ($n!$) duke përdorur funksionin `faktorieli`. Për tipin `int` të krijohet sinonimi `plote`, kurse për `double` të krijohet sinonimi `decimal`.

```
1  #include <iostream>
2  using namespace std;
3
4  double faktorieli(int n)
5  {
6      typedef int plote; //sinonim lokal
7      typedef double decimal; //sinonim lokal
8      plote i;
9      decimal F=1;
10
11     for (i=1;i<=n;i++)
12         F=F*i;
13     return F;
14 }
15
16 int main()
17 {
18     typedef int plote; //sinonim lokal
19     typedef double decimal; //sinonim lokal
20     plote n=0;
21     decimal fakt;
22
23     cout<<"Vlera n = ";
24     cin>>n;
25     fakt = faktorieli(n);
26     cout<< "Faktorieli F= "
27         << fakt<<"\n";
28     system("pause");
29     return 0;
30 }
```

Shembull 3.4

- Të shkruhet programi i cili përmes funksionit `vektori()` e llogarit prodhimin e anëtarëve negativ të vektorit $A(m)$. Për tipin `int` dhe `double` të krijohen sinonimet globale, `plote` respektivisht `decimal`.

$A(m) = \{-12, 3, 22, 6, -8, 33, -18, 4\}$

```
1  #include <iostream>
2  using namespace std;
3
4  typedef int plote; //sinonim global
5  typedef double decimal; //sinonim global
6
7  decimal vektori(decimal v[], plote m)
8  {
9      decimal p=1;
10     for (int i = 0; i < m; i++)
11         if (v[i] < 0)
12             {
13                 p=p* v[i];
14             }
15     return p;
16 }
17
18 int main()
19 {
20     const plote m=8;
21     plote i;
22     decimal A[m]={-12,3,22,6,-8,33,-18,4};
23
24     cout<<"Prodhimi i anëtarëve negativ S= "
25         << vektori(A,m)
26         <<endl;
27     system("pause");
28     return 0;
29 }
```

Numërimet (Grupimet)

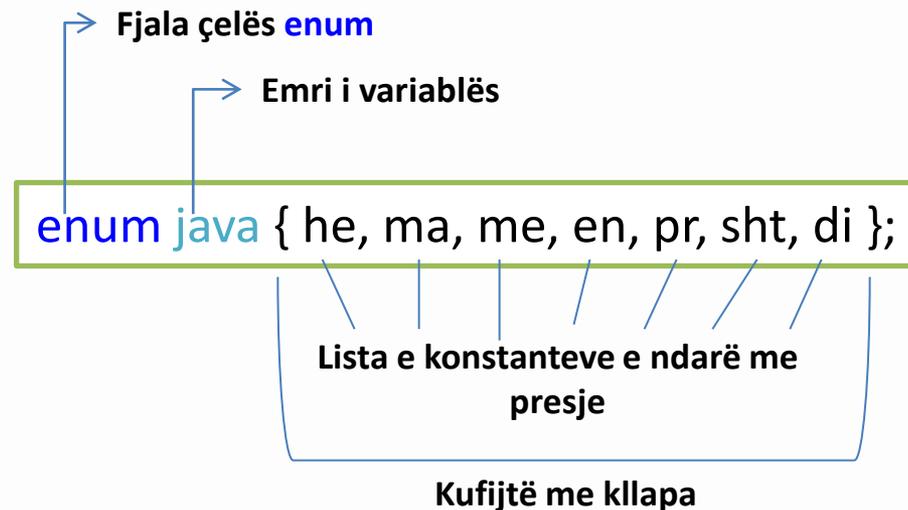
- Numërimi është një listë e konstanteve me të gjitha vlerat e mundshme.

{ kaltër, kuqe, portokalltë, gjelbër, zezë }

- Konstantet e emëruara ruhen në memorie si numra të plotë (integer) duke filluar nga 0, 1... (nëse nuk përcaktohet ndryshe)

Përcaktimi i grupit

- ❑ Një grup përcaktohet/definohet duke e shfrytëzuar komandën `enum`
- ❑ Anëtarët e grupit quhen edhe numërues (ang. enumerator)
- ❑ Sintaksa për deklarimin e një grupi të numëruar është:



Shembull 3.5

- Cilat nga 4 grupet e mëposhtme janë të përcaktuara saktë.

```
enum gradat {'A', 'B', 'C', 'D', 'F'};
```

```
enum gradat {A, B, C, D, F};
```

```
enum pozita {pari, 2ti, 3ti, 4ti};
```

```
enum pozita {pari, dyti, treti, katerti};
```

```
enum gradat {'A', 'B', 'C', 'D', 'F'};
```

```
enum gradat {A, B, C, D, F};
```

```
enum pozita {pari, 2ti, 3ti, 4ti};
```

```
enum pozita {pari, dyti, treti, katerti};
```



- Sa anëtarë ka grupi i mëposhtëm

```
enum qytetet {Prishtina, Prizreni, Tirana};
```

```
enum qytetet {Prishtina, Prizreni, Tirana};
```

3

- Sa është vlera e paracaktuar (default) për Tirana.

```
enum qytetet {Prishtina, Prizreni, Tirana};
```

```
enum qytetet {Prishtina, Prizreni, Tirana};
```

0

1

2

Shfrytëzimi i grupit

- Sintaksa e deklarimit të varablave të grupit

```
e v;
```

e – emri i grupit

v – variabla e deklaruar e grupit

- Shembull: për grupin e deklaruar **sportet**

```
enum sportet  
{  
    futboll,  
    basketboll,  
    volejboll,  
    karate,  
    boks  
};
```

```
sportet sportiPopullarizuar;  
sportiPopullarizuar=futboll;
```

Deklarimi i variablës sportiPopullarizuar dhe inicimi me vlerën **futboll**

Shembull 3.6

- Të krijohet grupi `shtetet` i cili përmban anëtarët: `Kosova`, `Shqiperia`, `Maqedonia`, `Gjermania` dhe `Suedia`. Më pas të deklarohet një variabël e tipit `shtetet` dhe të shtypet vlera e paracaktuar (default) për Gjermaninë.

```
1  #include <iostream>
2  using namespace std;
3
4  enum shtetet
5  {
6      Kosova,
7      Shqiperia,
8      Maqedonia,
9      Gjermania,
10     Suedia
11 };
12
13 int main()
14 {
15     shtetet shteti;
16     shteti = Gjermania;
17     cout << "\nGjermania e ka vleren e paracaktuar (default) = "
18          << shteti << endl ;
19     system("pause");
20     return 0;
21 }
```

Detyra: 3.1

- Të shkruhen 2 funksione me emrin `maksimumi()`. Njëri funksion të gjejë dhe ta shtyp numrin më të madh në vektorin $A(m)$, kurse funksioni tjetër numrin më të madh në matricën $B(m,m)$. Vektori dhe matrica gjatë deklarimit të iniciohen me vlera fillestare.

Detyra: 3.2

- Të shkruhen 3 funksione me emrin **siperfaqja()** të cilat llogarisin sipërfaqen e **kubit**, **sferës** dhe **konit**. Funksionet të thirren për vlera të lexuara përmes tastierës.

Detyra: 3.3

- Të shkruhet programi i cili përmban një funksion të mbingarkuar i cili shtyp vlerat e secilit parametër. Njëri funksion të ketë dy, kurse tjetri tre parametra të tipit të çfarëdoshëm. Për tipat e të dhënave të krijohen sinonime.

Detyra: 3.4

- Të shkruhet programi në të cilin është i deklaruar një grup me emrin **veturat**. Grupimi të përmbajë disa lloje të veturave dhe më pas të shtypet vlera e paracaktuar për veturën e preferuar.

Detyra: 3.5

- Të shkruhet programi në të cilin është i deklaruar një grup me emrin **stinet** i cili përmban katër stinët e vitit. Më pas të shtypen vlerat e paracaktuara (default) për secilën stinë të vitit.



Pyetje ?