



Azerbaijan Computing Olympiad

Hazırladı: Kanan Ağarlı

MÖVZU 02

Data tiplər

- C++ proqramlaşdırma dilində dəyişənləri təyin edərkən aşağıdakı cədvəldə göstərilən data tiplərdən istifadə etmək mütləqdir:

Data tip	Təyinatı	Yaddaşı	Interval
int	Tam ədədləri təyin etmək üçündür	32 bit (4 bayt)	$[-2^{31}, 2^{31} - 1]$
long long	Tam ədədləri təyin etmək üçündür	64 bit (8 bayt)	$[-2^{63}, 2^{63} - 1]$
double	Həqiqi (kəsr) ədədləri təyin etmək üçündür	64 bit (8 bayt)	$[-10^{308}, 10^{308}]$
char	Simvol dəyərləri təyin etmək üçündür	8 bit (1 bayt)	$[0, 255]$
bool	Məntiqi (true/false) dəyərləri təyin etmək üçündür	8 bit (1 bayt)	$[false, true]$ və ya $[0, 1]$

Dəyişənlər

- Dəyişənlər dəyərləri saxlamaq üçün istifadə olunur. Siz riyaziyyat fənnində dəyişənlərlə tanış olmuşunuz (məs: x, y, a, b, c və s.). Proqramlaşdırmada da belə dəyişən adlarından istifadə edə bilərik. Nümunə üçün x = 5 dəyişəni təyin edək və x-in qiymətini çıxışa verək:

<pre>1 #include <bits/stdc++.h> 2 using namespace std; 3 int main() 4 { 5 int x = 5; 6 cout<<x<<endl; 7 }</pre>	Çıxış: 5
---	-------------

- Gördüyünüz kimi x dəyişənini təyin edərkən onun qarşısında tam ədədi bildirən data tipini (int) təyin etdik. Dəyişənlərə qiymətləri sonradan da vermək, lakin dəyişənləri istifadə etməzdən əvvəl mütləq qiymətləri təyin olunmalıdır. Aşağıdakı nümunəyə baxaq:

<pre>1 #include <bits/stdc++.h> 2 using namespace std; 3 int main() 4 { 5 int x; 6 x = 5; 7 cout<<x<<endl; 8 }</pre>	Çıxış: 5
--	-------------

➤ İndi isə gəlin yuxarıdakı cədvəldə göstərilən bütün data tiplərə aid nümunəyə baxaq:

1	#include <bits/stdc++.h>	Çıxış:
2	using namespace std;	1000000
3	int main()	10000000000000
4	{	24.578
5	int a = 1000000;	A
6	long long b = 10000000000000;	1
7	double c = 24.578;	
8	char d = 'A';	
9	bool e = true;	
10	cout<<a<<endl;	
11	cout<<b<<endl;	
12	cout<<c<<endl;	
13	cout<<d<<endl;	
14	cout<<e<<endl;	
15	}	

➤ Dəyişənlərin adlarını təyin edərkən aşağıdakı qaydalara əməl etmək lazımdır:

- Bütün dəyişənlərin adları fərqli olmalıdır.
- Dəyişənləri adlandırarkən qısa adlardan (məs: x, y və s.) və ya daha detallı adlardan (məs: age, sum, total, name və s.) istifadə etmək olar.
- Daha detallı adlardan istifadə olunması tövsiyyə olunandır. Çünki bu halda yazdığımız kodlar daha yaxşı başa düşülən olur.
- Adların tərkibində yalnız ingilis latın hərfləri, rəqəmlər, alt-xətt (underscore _) və valyuta işarələri (currency characters - \$) ola bilər, lakin ilk simvolu rəqəm ola bilməz.
- Adların hərflərinin böyük və kiçik olmasına diqqət etmək lazımdır. Məsələn x və X dəyişənləri fərqli dəyişənlər hesab edilir.
- Adların tərkibində boşluqlardan (space) və digər xüsusi simvollar (!, #, % və s.) istifadə etmək olmaz.
- Açar sözlər (reserved və ya key words) dəyişənlərin adı kimi istifadə oluna bilməz (məs: **int**, **main**, **include**, **std** və s. kimi açar sözləri dəyişənin adı kimi istifadə edə bilmərik).

```

1  #include <bits/stdc++.h>
2  using namespace std;
3  int main()
4  {
5      int age = 14;
6      int x1 = 5, x2 = 10;
7      double area = 21.5;
8  }
```

Cəbri operatorlar

- Proqramlaşdırmada cəbri əməliyyatlardan tez-tez istifadə olunur. Cəbri əməliyyatları yerinə yetirən əsas operatorlar aşağıdakı cədvəldə göstərilib:

Operator	Təyinatı
+	Toplama
-	Çıxma
*	Vurma
/	Bölmə
%	Qalıq

- Cəbri operatorlara aid aşağıdakı nümunəyə baxaq:

1	#include <bits/stdc++.h>	Çıxış:
2	using namespace std;	14
3	int main()	8
4	{	33
5	int x = 11, y = 3;	3
6	cout<<x + y<<endl;	2
7	cout<<x - y<<endl;	
8	cout<<x * y<<endl;	
9	cout<<x / y<<endl;	
10	cout<<x % y<<endl;	
11	}	

- C++ dilində kəsrlı bölmə üçün xüsusi operator yoxdur. Kəsrlı bölməni hesablamaq üçün dəyişənləri **double** tipində təyin etmək lazımdır və ya hesablama zamanı dəyişənin qarşısında **double** qeyd edə bilərik:

1	#include <bits/stdc++.h>	Çıxış:
2	using namespace std;	3.66667
3	int main()	
4	{	
5	double x = 11, y = 3;	
6	cout<<x / y<<endl;	
7	}	

1	#include <bits/stdc++.h>	Çıxış:
2	using namespace std;	3.66667
3	int main()	
4	{	
5	int x = 11, y = 3;	
6	cout<<(double)x / y<<endl;	
7	}	

Mənimləmə operatorları

- Proqramlaşdırmada mənimləmə operatoru '=' operatorudur. Bu operatoru riyaziyyatdakı bərabərlik işarəsi ilə eyniləşdirmək doğru deyil. Aşağıdakı kod nümunəsinə nəzər salaq:

1	#include <bits/stdc++.h>	Çıxış:
2	using namespace std;	5
3	int main()	7
4	{	
5	int x = 5;	
6	cout<<x<<endl;	
7	x = x + 2;	
8	cout<<x<<endl;	
9	}	

- Yuxarıdakı nümunə kodun 5-ci sətirində x dəyişəninə 5 qiymətini verdik və 6-cı sətirdə dəyərini çıxışa verdik. 7-ci sətirdə isə x dəyişəninə üzərinə 2 əlavə etdik (sağ tərəf öncə hesablanır və alınan nəticə sol tərəfə verilir) və 8-ci sətirdə dəyərini çıxışa verdik. Bu əməliyyatı qısa formada operatorları birləşdirərək də yazmaq olar:

1	#include <bits/stdc++.h>	Çıxış:
2	using namespace std;	5
3	int main()	7
4	{	
5	int x = 5;	
6	cout<<x<<endl;	
7	x += 2;	
8	cout<<x<<endl;	
9	}	

Artma/Azalma (increment/decrement) operatorları

- Dəyişənlərin qiymətini 1 vahid artırmaq və ya azaltmaq üçün uyğun olaraq ++ və -- operatorından istifadə edə bilərik:

1	#include <bits/stdc++.h>	Çıxış:
2	using namespace std;	6 3
3	int main()	
4	{	
5	int x = 5, y = 4;	
6	x++; //x += 1, x = x + 1	
7	y--; //y -= 1, y = y - 1	
8	cout<<x<<" "<<y<<endl;	
9	}	

Giriş (input)

- Biz artıq çıxış əməliyyatını icra etməyi bilirik və bu əməliyyat **cout<<** şəklində yerinə yetirilir. İndi isə gəlin giriş əməliyyatını öyrənək. Proqramda dəyişənlərə istifadəçi tərəfindən qiymətlərin verilməsi üçün giriş (input) əməliyyatı icra olunmalıdır. C++ dilində giriş əmri **cin>>** əməliyyatı ilə yerinə yetirilir. Aşağıdakı kod nümunəsinə baxaq:

1	<code>#include <bits/stdc++.h></code>	Çıxış:
2	<code>using namespace std;</code>	4 5
3	<code>int main()</code>	9
4	<code>{</code>	
5	<code> int x, y;</code>	
6	<code> cin>>x>>y;</code>	
7	<code> cout<<x + y<<endl;</code>	
8	<code>}</code>	

- Bu nümunənin 5-ci sətirində x və y dəyişənləri təyin olunub. 6-cı sətirdə isə həmin dəyişənlərə istifadəçi tərəfindən qiymətlər verilir (nümunə üçün x = 4 və y = 5 qiymətləri verilir). 7-ci sətirdə x + y cəmi çıxışa verilir.

Tapşırıqlar

- azeco.eolymp.space : 20 məsələ (nömrələri: 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25)