



Rıdvan Tülemen
Artun Çolak
Yunus Emre Bulut

C++

Aşağıdaki kodlardaki fark '**std**'nin kullanılmasıdır. İkinci kodda std kullanılmamasındaki amaç aynı isimdeki method ve class isimlerinin olası çakışmalarından kaçınılmasıdır.

- Cin: input alır (scanf)
- Cout: output verir (printf)

main.cpp X

main.cpp > main()

```
1  #include <iostream>
2
3  using namespace std;
4  int main(){
5      int input1;
6      cin >> input1;
7      cout << input1;
8  }
```

main.cpp X

main.cpp > ...

```
1  #include <iostream>
2
3  int main(){
4      int input1;
5      std::cin >> input1;
6      std::cout << input1;
7  }
```

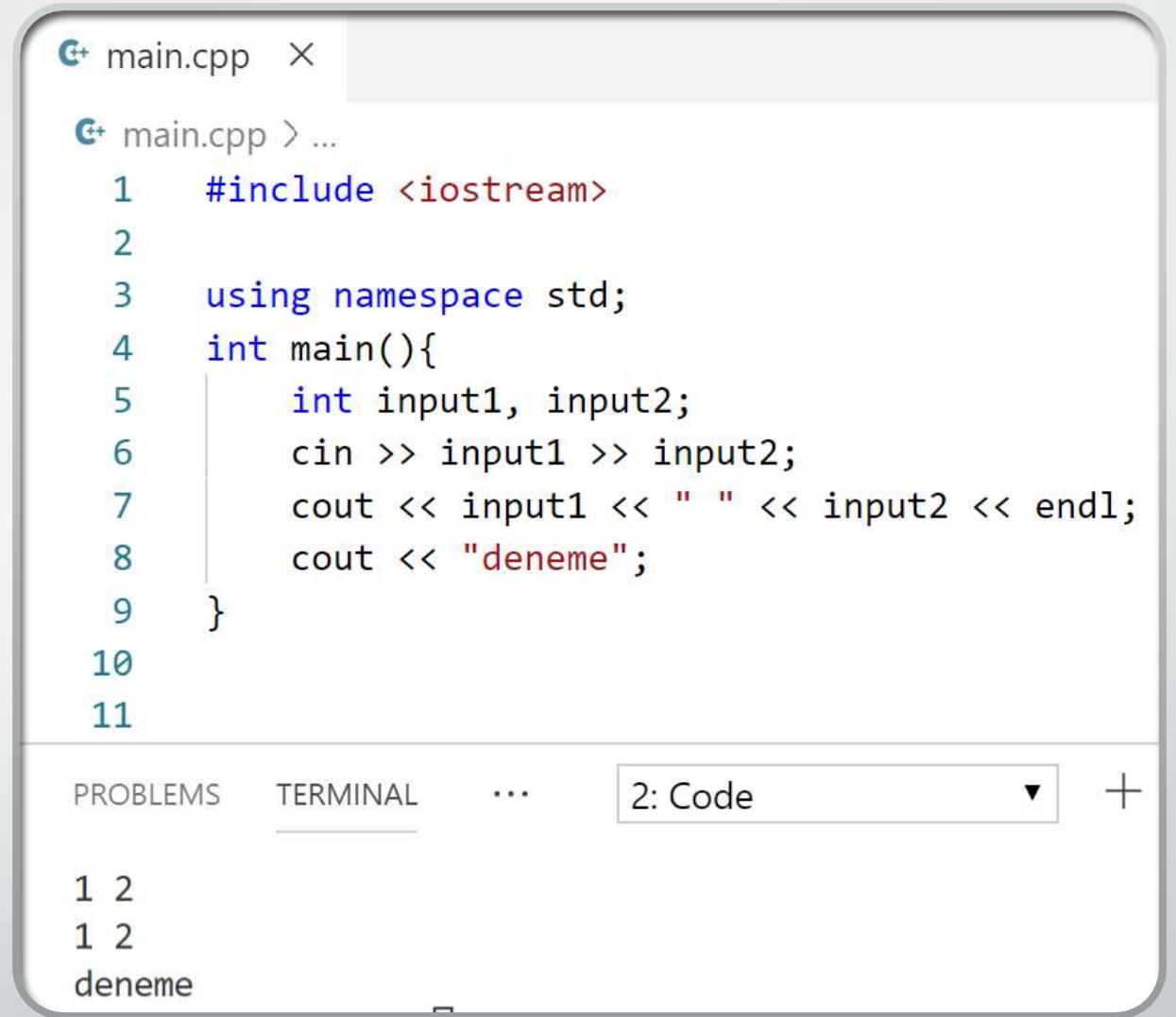
C++ içinde C
kodlarını kullanabilirsiniz.

main.cpp X

main.cpp > ...

```
1  #include <iostream>
2  #include <stdio.h>
3
4  using namespace std;
5  int main(){
6      int input1;
7      cin >> input1;
8      cout << input1;
9
10     scanf("%d", &input1);
11     printf("%d", input1);
12 }
```

Birden
fazla input içeren küçük
bir program



```
main.cpp X
main.cpp > ...
1  #include <iostream>
2
3  using namespace std;
4  int main(){
5      int input1, input2;
6      cin >> input1 >> input2;
7      cout << input1 << " " << input2 << endl;
8      cout << "deneme";
9  }
10
11
```

PROBLEMS TERMINAL ... 2: Code ▼ +

```
1 2
1 2
deneme
```

Vector kütüphanesi

Vector, *array* benzeri bir yapıdır. Array'den farkı *binary tree* mantığını kullanan bir *linked list* olmasıdır. Yani işlemlerimiz array'den daha hızlı çalışacaktır. Ayrıca sonradan boyutu değiştirilebilir.

main.cpp

main()

```
1  #include <iostream>
2  #include <vector>
3
4  using namespace std;
5  int main(){
6      int input1, input2;
7      vector<int> list;
8
9      cin >> input1 >> input2;
10     list.push_back(input1);
11     list.push_back(input2);
12
13     cout << list.at(0) << " " << list.at(1) << endl;
14     cout << list[0] << " " << list[1] << endl;
15 }
16
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

2: Code

1 2

1 2

1 2


2D vector tanımlamak

```
main.cpp ×  
main.cpp > main()  
1  #include <iostream>  
2  #include <vector>  
3  
4  using namespace std;  
5  int main(){  
6      int N, M;  
7      cin >> N >> M;  
8  
9      vector<vector<int>> list(N, vector<int>(M));  
10  
11     for(int i = 0; i < N; ++i){  
12         for(int j = 0; j < M; ++j){  
13             int input;  
14             cin >> input;  
15             list[i][j] = input;  
16         }  
17     }  
18 }
```

1. Sınav 2. Soru

| | | |
|----|---|----|
| 3 | 5 | 4 |
| 12 | 8 | 7 |
| 4 | 9 | 10 |

| | | | |
|---|----|---|----|
| 2 | 5 | 3 | 6 |
| 6 | 7 | 8 | 9 |
| 4 | 13 | 4 | 1 |
| 3 | 8 | 2 | 11 |



| | | | | |
|----|----|---|----|---|
| 0 | 2 | 3 | 3 | 4 |
| 5 | 7 | 5 | 11 | 9 |
| 4 | 17 | 3 | 3 | 8 |
| 12 | 3 | 0 | 6 | 4 |
| 8 | 16 | 1 | 3 | 3 |

Bu örnekte cevap
72'dir.

Soru1:

Rakamları toplamı asal sayı yapan ve son rakamı hariç tüm rakamları toplamı son rakamın bir fazlasına eşit olan sayılara ***zipçikti sayılar*** denir. Klavyeden girilen sayıya kadar olan tüm zipçikti sayıları yazdırınız.

- Örnek 1: 122; $1 + 2 + 2 = 5 \rightarrow$ asal, $1 + 2 = 2 + 1$
- Örnek 2: 403; $4 + 0 + 3 = 7 \rightarrow$ asal, $4 + 0 = 3 + 1$

Soru 2:

Size verilen $N \times M$ matristeki tüm karelerin sayısını bulunuz.

Örnek olarak $N = 2$ ve $M = 3$ iken, toplam kare sayısı **8**'dir.

Kareler: {1}, {2}, {3}, {4}, {5}, {6}

{1, 2, 4, 5}, {2, 3, 5, 6}

| | | |
|---|---|---|
| 1 | 2 | 3 |
| 4 | 5 | 6 |