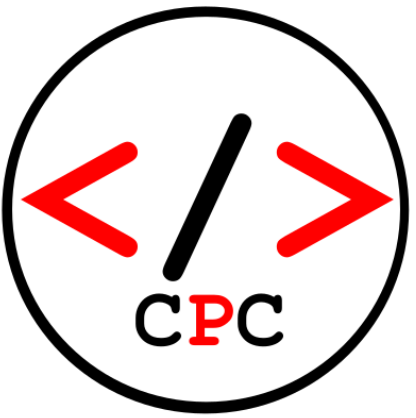


# Pendahuluan Competitive Programming (CP)

Rosa A. S.

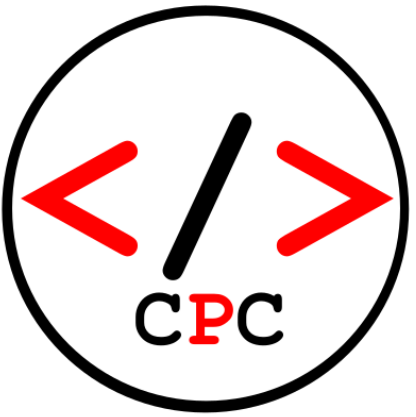
# Competitive **Programming** Club – Computer **Science** UPI





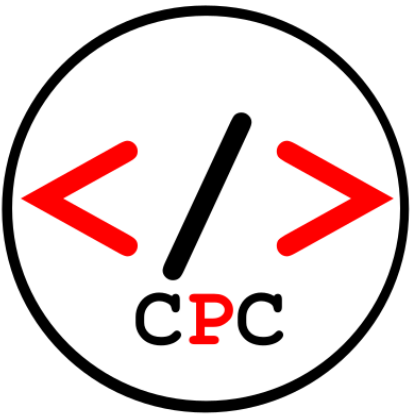
- Orang hebat karena melakukan hal yang hebat
- Tidak ada orang jadi hebat karena melakukan hal yang biasa saja dan mudah
- Kalau mudah ya akan banyak orang yang melakukan hal itu, akhirnya menjadi biasa
- Kemauan untuk eksplorasi – rasa penasaran yang diwujudkan dengan baik





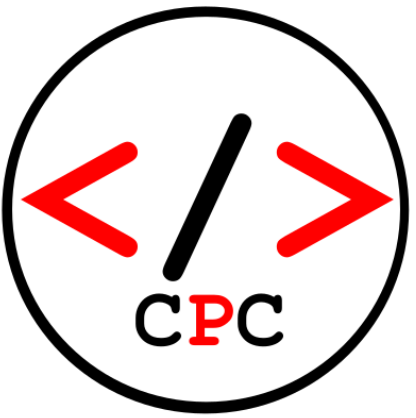
# Apa itu Competitive Programming?

- Competitive Programming adalah sebuah kegiatan penyelesaian masalah (*problem solving*) tertentu menggunakan program komputer dengan mengedepankan algoritma yang paling optimal.
- Hal yang perlu dikuasai:
  - Kecepatan jalannya program
  - Pemahaman Algoritma
  - Kreativitas
  - Kemampuan Pemrograman (*Programming Skill*)
  - Dasar Matematika
  - Kemampuan melakukan perbaikan program (*Debugging*)



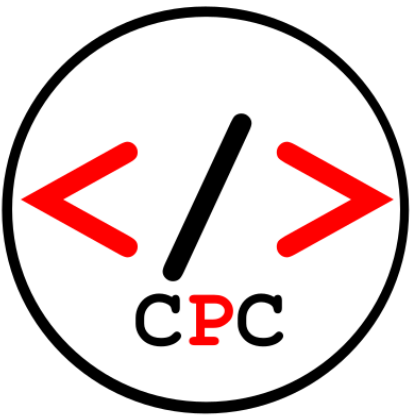
## Manfaat CP

- Melatih kecepatan dan fokus
- Melatih kesiapan menghadapi sesuatu (jangan dibuat beban, tapi pengisi waktu luang)
- Kerjasama Tim
- Kemampuan yang banyak dilirik perusahaan



# Kebutuhan Perusahaan

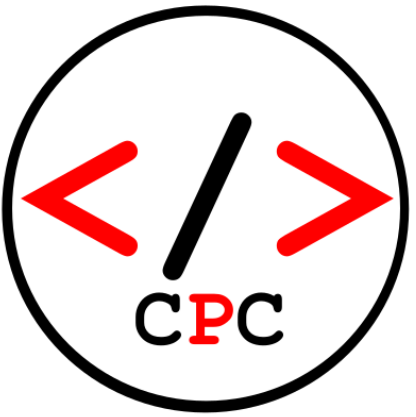
- Perusahaan membutuhkan orang yang cerdas (tajam pikiran)
- Kemampuan menganalisis permasalahan yang baik sesuai bidangnya
- Kemampuan bekerja yang baik (*team work*, komunikasi, motivasi, dll)
- Menyelesaikan sesuatu dengan algoritma yang cepat
- Terbiasa dengan program yang tidak banyak kesalahan (*error/bug*)
- Terbiasa melakukan perbaikan kesalahan (*debugging*)



# Tips Menguasai CP

- Memahami Konsep Algoritma
- Mulai berlatih dari kasus/soal yang mudah
- Berlatih secara berkelanjutan (bukan hanya semangat di awal saja)
- Mempelajari mengoptimalkan program
- Mengikuti Kompetisi (mengikuti dan kalah, lebih terhormat dibanding tidak berani ikut sama sekali)
- Selalu menjadi jiwa yang dinamis dan *up-to-date*

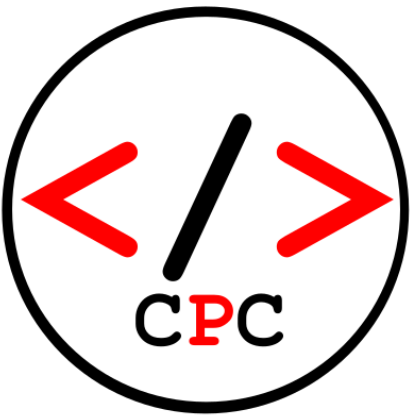




# Proses dalam Menyelesaikan Soal CP

- Desain Algoritma
  - Konsep apa yang digunakan (Teknik-teknik Algoritma)
  - Efisiensi
  - Cepat
- Implementasi Algoritma
  - Menguasai bahasa pemrograman
  - Dapat mengimplementasikan desain algoritma menjadi kode program





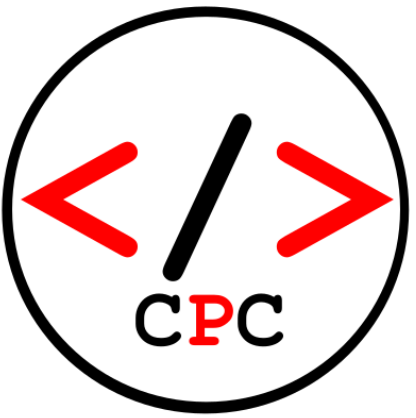
# Platform Latihan CP

- TLX - <https://tlx.toki.id/> - berbahasa Indonesia
- TopCoder – berbahasa Inggris
- Hackerrank – berbahasa Inggris
- UVa Online Judge - <https://onlinejudge.org/> – berbahasa Inggris
- dll



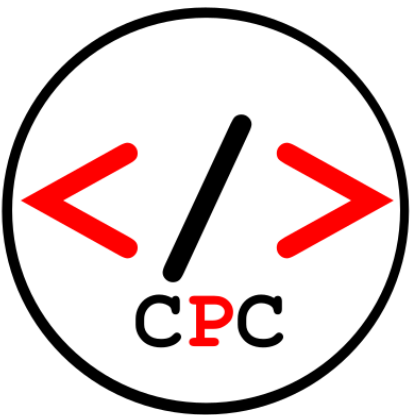
# Tahapan Lomba CP

- *Warming Up* atau pemanasan
  - Tahapan ini dimaksudkan agar peserta “berkenalan” dengan mesin dan *server* yang akan digunakan untuk lomba. Biasanya akan diberikan beberapa soal untuk dicoba diselesaikan dan diunggah ke *server*. Pemanasan biasanya dilakukan sehari sebelum babak penyisihan.
- Babak Penyisihan secara *online*
  - Tahapan ini biasanya dilakukan untuk menyaring 15-30 besar dari tim yang bertanding untuk maju ke babak final. Tahapan ini biasanya dilakukan secara *online* di tempat masing-masing.
- Babak Final
  - Tahap ini biasanya dilakukan secara *onsite* di tempat penyelenggara lomba dan diikuti 15-30 besar tim yang berhasil disaring dari babak penyisihan. Biasanya setiap tim hanya diberikan satu buah komputer beserta alat tulis dan beberapa lembar kertas untuk menyelesaikan soal-soal pemrograman yang diberikan.



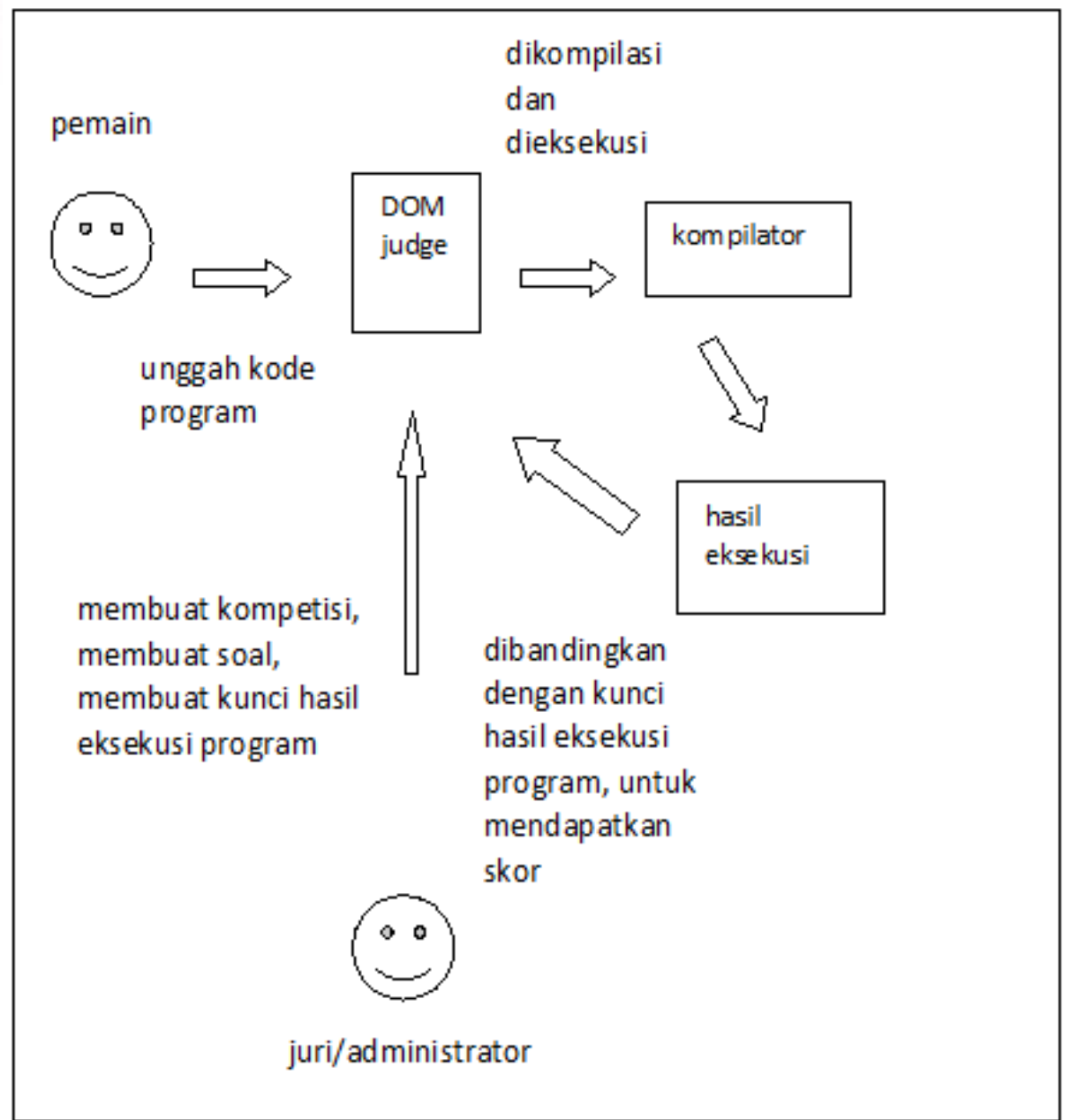
# Klarifikasi pada CP

- Waktu bertanya terkait soal biasanya 1-2 jam pertama kompetisi
- Pertanyaan yang diijinkan biasanya dengan jawaban yes no
- Tidak bisa bertanya yang salah test case yang mana hehe 😁
  - Buat sendiri test case yang sekiranya menjadi test case yang diujikan (*hidden case*), misalnya
    - Jika data masukan banyak
    - Jika data masukan sedikit
    - Jika data masukan bernilai besar sesuai batasan paling atas soal (*corner case*)
    - Jika data masukan bernilai kecil atau negatif bergantung pada batasan paling kecil dari soal (*corner case*)

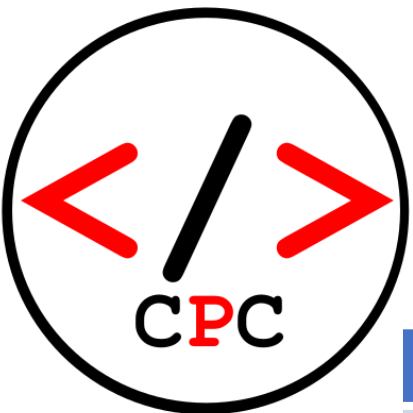


# Online Judge

Server biasanya based on Linux







# Verdict/Putusan Online Judge

Verdict/Putusan	Keterangan
<b>CORRECT</b> atau <b>ACCEPTED (ACC)</b>	Jawaban benar dan jawaban diterima, memenuhi batas waktu dan memori yang telah ditentukan pada soal
<b>COMPILER-ERROR</b> atau <b>COMPILATION ERROR</b>	Gagal kompilasi
<b>TIMELIMIT</b> atau <b>TIME LIMIT EXCEDED (TLE)</b>	Waktu eksekusi melebihi waktu yang ditentukan



# Verdict/Putusan Online Judge

Verdict/Putusan	Keterangan
<b>RUN-ERROR</b>  atau  <b>RUNTIME ERROR</b>	<ul style="list-style-type: none"><li>• pembagian dengan 0</li><li>• nilai <i>return</i> bukan 0</li><li>• akses <i>array</i> lebih dari alokasi</li><li>• penggunaan memori lebih dari yang ditentukan</li><li>• penggunaan komentar (sebaiknya tanpa komentar karena ada mesin yang tidak menerima komentar)</li><li>• tidak membuat variabel yang tidak pernah digunakan</li><li>• usahakan tidak membuat fungsi dan prosedur sendiri (karena ada mesin yang tidak menerima fungsi atau prosedur sendiri)</li><li>• pembuatan <i>array</i> dengan jumlah elemen terlalu banyak, misal</li><li>• <code>int arr[10000][10000]</code></li></ul>



# Verdict/Putusan Online Judge

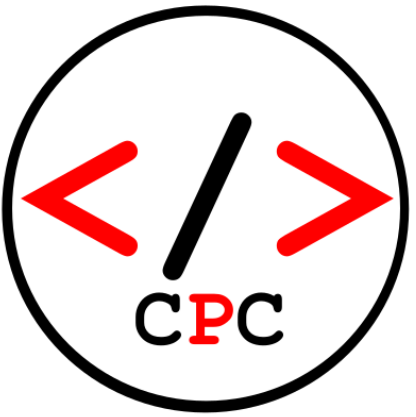
Verdict/Putusan	Keterangan
<b>NO-OUTPUT</b>	tidak ada yang dihasilkan sebagai <i>output</i> lupa menuliskan kode sebagai <i>output</i>
<b>WRONG-ANSWER (WA)</b>  atau <b>PRESENTATION-ERROR</b>	keluaran atau hasil keluaran dari program tidak sesuai permintaan soal  juga dapat disebabkan kelebihan spasi atau <i>newline</i> terutama jika <i>verdict</i> yang muncul adalah PRESENTATION-ERROR
<b>TOO-LATE</b>	waktu kompetisi sudah habis



# Penilaian CP

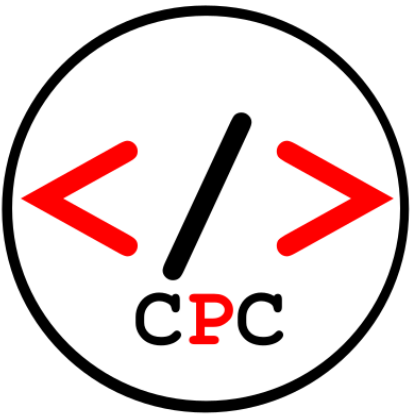
Nilai	Waktu	Soal A	Soal B	Soal C	Soal D	Soal E
4	259	2 (39 + 20)	2 (120 + 20)	1 (37 + 0)	0	1 (23 + 0)
4	260	1 (11 + 0)	4 (139 + 60)	1 (36 + 0)	1	1 (14 + 0)





# Beberapa Tipe Menghadapi Soal CP

1. belum pernah melihat tipe soal
2. sudah pernah melihat tipe soal namun belum dapat menyelesaikannya
3. sudah pernah menyelesaikan tipe soal ini sebelumnya



# Karakteristik Soal CP

- Soal terdefinisi dengan jelas terkait Batasan yang diberikan terkait waktu eksekusi yang diijinkan, penggunaan memori yang diijinkan
- Batasan masukan dan keluaran jelas beserta batasannya
- Diselesaikan dengan membuat program computer berbasis command prompt/console



# Eksekusi

```
g++ namafile.cpp -o namafile
```

```
g++ -std=c++20 -O2 -Wall namafile.cpp -o namafile
```

```
namafile < input.txt
```



# Template Kode Program

```
#include <bits/stdc++.h>

using namespace std;

int main(){
    ios::sync_with_stdio(0);
    cin.tie(0);

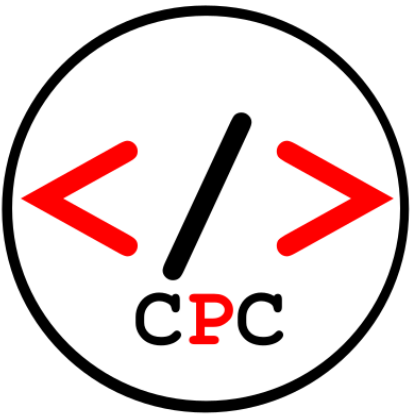
    //solusi
    return 0;
}
```





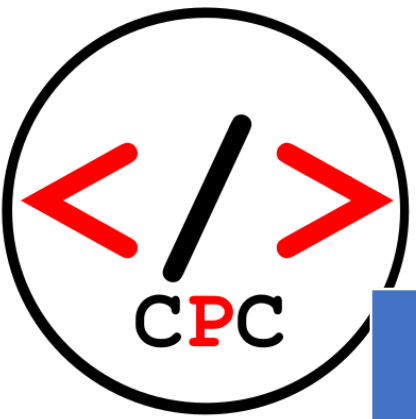
# Input / Output

C Input	C++ Input
<pre>int a, b; scanf("%d %d", &amp;a, &amp;b);</pre>	<pre>int a, b; string x; cin &gt;&gt; a &gt;&gt; b &gt;&gt; x;</pre> <p>menerima masukan berupa satu baris string</p> <pre>string s; getline(cin, s);</pre>
	<p>menerima masukan beberapa data yang tidak diketahui jumlahnya</p> <pre>while (cin &gt;&gt; s) {     //code }</pre>
	<pre>while (cin &gt;&gt; s &gt;&gt; x) {     //code }</pre>
Output	Output
<pre>int a = 99, b = 19; printf("%d %d\n", a, b);</pre>	<pre>int a = 99, b = 19; string x = "competitive"; cout &lt;&lt; a &lt;&lt; " " &lt;&lt; b &lt;&lt; " " &lt;&lt; x &lt;&lt; "\n";</pre>



# Tipe Data Bilangan Besar

- Bilangan riil gunakan
  - double
  - long double
- Bilangan bulat gunakan
  - long long
  - unsigned long long
- Makro
  - `typedef long long ll;`



# Operator

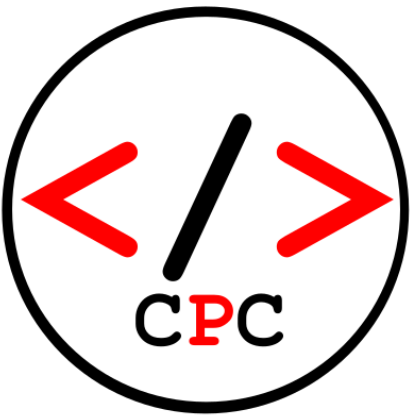
Operator	Example	Equivalent to
=	a = b;	a = b;
+=	a += b;	a = a + b;
-=	a -= b;	a = a - b;
*=	a *= b;	a = a * b;
/=	a /= b;	a = a / b;
%=	a %= b;	a = a % b;



# Tipe masukan

- Satu kata
- Satu baris string
- Berbaris-baris string
- Barbaris-baris string dan di-split
- Satu Integer
- Banyak Integer





# Satu Kata

```
#include <bits/stdc++.h>

using namespace std;

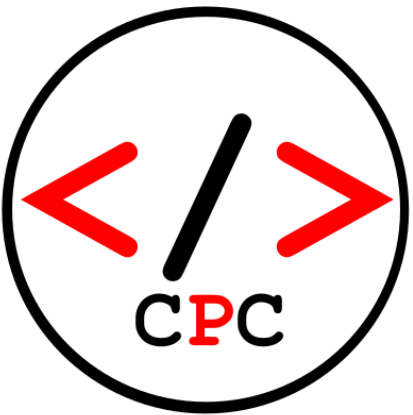
int main(){
    ios::sync_with_stdio(0);
    cin.tie(0);

    string str;

    cin >> str;

    cout << str;

    return 0;
}
```



# Satu Baris String

```
#include <bits/stdc++.h>

using namespace std;

int main(){

    ios::sync_with_stdio(0);
    cin.tie(0);

    string str;

    getline(cin, str);

    cout << str << "\n";

    return 0;
}
```



# Banyak Baris String

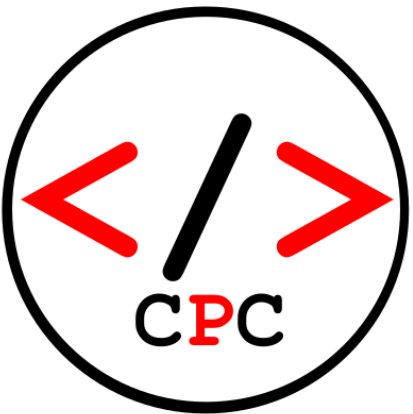
```
#include <bits/stdc++.h>

using namespace std;

int main() {
    ios::sync_with_stdio(0);
    cin.tie();

    string str;
    while(getline(cin, str)) {
        cout << str << "\n";
    }

    return 0;
}
```



# Split String

```
#include <bits/stdc++.h>

using namespace std;

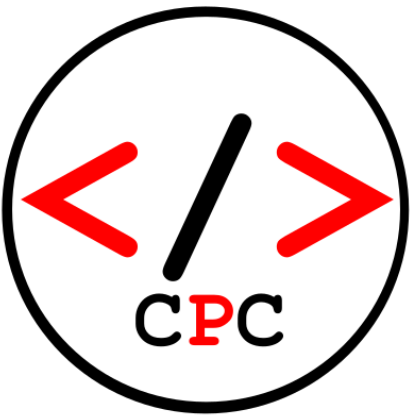
int main(){

    string s;

    while(getline(cin, s)){
        stringstream ss(s);
        string word;
        while (ss >> word) {
            cout << word << "\n";
        }
    }

    return 0;
}
```





# Satu Integer

```
#include <bits/stdc++.h>

using namespace std;

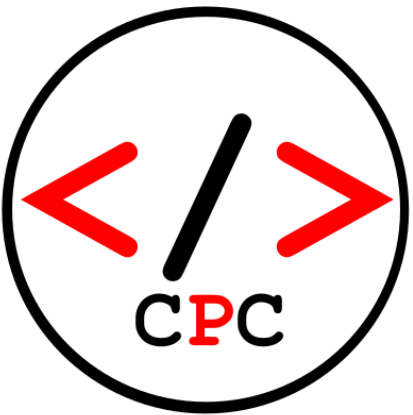
int main(){
    ios::sync_with_stdio(0);
    cin.tie(0);

    long long a;

    cin >> a;

    cout << a << "\n";

    return 0;
}
```



# Banyak Integer

```
#include <bits/stdc++.h>

using namespace std;

int main(){
    ios::sync_with_stdio(0);
    cin.tie(0);

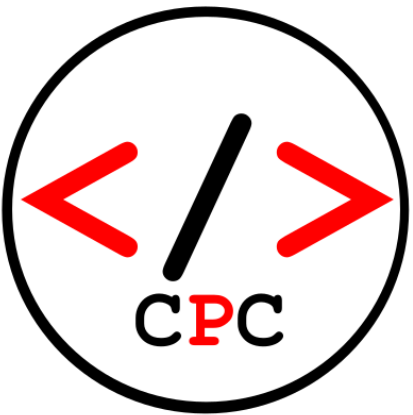
    long long a;
    while(cin >> a){
        cout << a << "\n";
    }

    return 0;
}
```



# Ternary IF

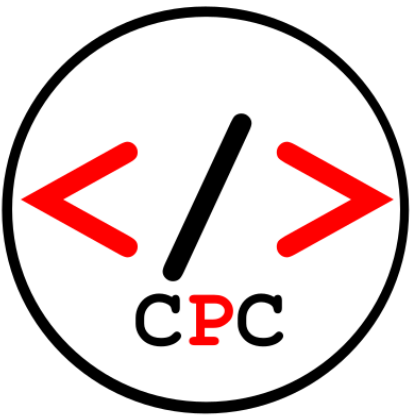
- `hasil = (a<0) ? "Bilangan negatif" : "Bilangan positif";`
- `a<b ? a : b;`
- `return( a < b ? a : b );`



**IF**

```
if(c > 0) cout << "Kembalian\n";  
else if(c < 0) cout << "Kurang\n";  
else cout << "Pas\n";
```





## Next: Konversi

- Integer ke string
- String ke Integer
- Double ke string
- String ke double
- Long long ke string
- String ke long long