

Funciones inline

```
#include <iostream>
using namespace std;

inline float square(const float x) {return x*x;}
void some_other_function(char broiled, float other_stuff);
float calc_iq(const float hatsize, const int age, const float num_feet);

int main()
{
    float y = square(6.2);

    return 0;
}
```

```
#include <iostream>
using namespace std;
```


```
inline float square(const float x) {return x*x;}
void some_other_function(char broiled, float other_stuff);
float calc_iq(const float hatsize, const int age, const float num_feet);

int main()
{
    float y = square(6.2);

    return 0;
}
```

```
#include <iostream>
using namespace std;
```

```
inline float square(const float x) {return x*x;}
void some_other_function(char broiled, float other_stuff);
float calc_iq( const float hatsize, const int age, const float num_feet);
```

<pre>int main() { float y = square(6.2); return 0; }</pre>		<pre>int main() { float y = 6.2 * 6.2; return 0; }</pre>
--	---	--

Cuándo utilizar **inline**

- Sólo compensa si la función es muy pequeña (muy pocas sentencias)
- Elimina el coste de efectuar una llamada a función
- Cuando la función sea llamada un número pequeño de veces, el incremento de velocidad de ejecución no será perceptible