

# Algorists Template for Talks

Ulises Tirado Zatarain

Nov, 2020

- Example of math using an underscore and mathcal:  $\alpha_i \in \mathcal{A}$
- Example of math using an arrow and mathbb:  $\vec{\beta} \notin \mathbb{B}$
- Example of math using an super script and mathbf:  $\mathbf{c}^k \subset \mathbf{C}$
- Following there is a more complicated formula:

$$\Gamma^n = \int_a^b \frac{4py}{k\epsilon} dy \times \begin{bmatrix} x & y & z \\ u & v & w \end{bmatrix}$$

- Example of math using an underscore and mathcal:  $a_i \in \mathcal{A}$
- Example of math using an arrow and mathbb:  $\vec{\beta} \notin \mathbb{B}$
- Example of math using an super script and mathbf:  $\mathbf{c}^k \subset \mathbb{C}$
- Following there is a more complicated formula:

$$\Gamma^n = \int_a^b \frac{4py}{k\epsilon} dy \times \begin{bmatrix} x & y & z \\ u & v & w \end{bmatrix}$$

- Example of math using an underscore and mathcal:  $a_i \in \mathcal{A}$
- Example of math using an arrow and mathbb:  $\vec{\beta} \notin \mathbb{B}$
- Example of math using an super script and mathbf:  $c^k \subset \mathbf{C}$
- Following there is a more complicated formula:

$$\Gamma^n = \int_a^b \frac{4py}{k\epsilon} dy \times \begin{bmatrix} x & y & z \\ u & v & w \end{bmatrix}$$

- Example of math using an underscore and mathcal:  $a_i \in \mathcal{A}$
- Example of math using an arrow and mathbb:  $\vec{\beta} \notin \mathbb{B}$
- Example of math using an super script and mathbf:  $c^k \subset \mathbf{C}$
- Following there is a more complicated formula:

$$\Gamma^n = \int_a^b \frac{4py}{k\epsilon} dy \times \begin{bmatrix} x & y & z \\ u & v & w \end{bmatrix}$$

Let see what the code looks like:

```
for (auto& x: S) {  
    if (x > 10) {  
        printf("%d\n", x);  
    }  
}
```

# Including source code files

You can also include a cpp file:

```
#include <iostream>
```

```
int function(int& x) {  
    std::cout << "x_=" << ++x << std::endl;  
    return x;  
}
```

```
int main(int argc, char const *argv[]) {  
    int p = 4, q = 7;  
    int y = p < q ? function(p) : function(q);  
    std::cout << "y_=" << y << std::endl;  
    return 0;  
}
```

---

---

```
1 while  $n > 5$  do:
2   |   instructions;
3   if condition then
4     |   instructions1;
5     |   instructions2;
6   else:
7     |   instructions3;
8   end
9 end
```

---



## Algorithm Example (pseudocode)

```
map of <string , integer> dictionary;  
vector of integer numbers;  
pair of <integer , integer> position;  
set of integer S;  
unique pointer of character c = null;  
const reference  
for each any x in S do:  
    if x > 10 then:  
        writeln("%d" , x);  
    end  
end  
  
integer index = 0;  
repeat:  
    index++;  
until index >= 10;
```