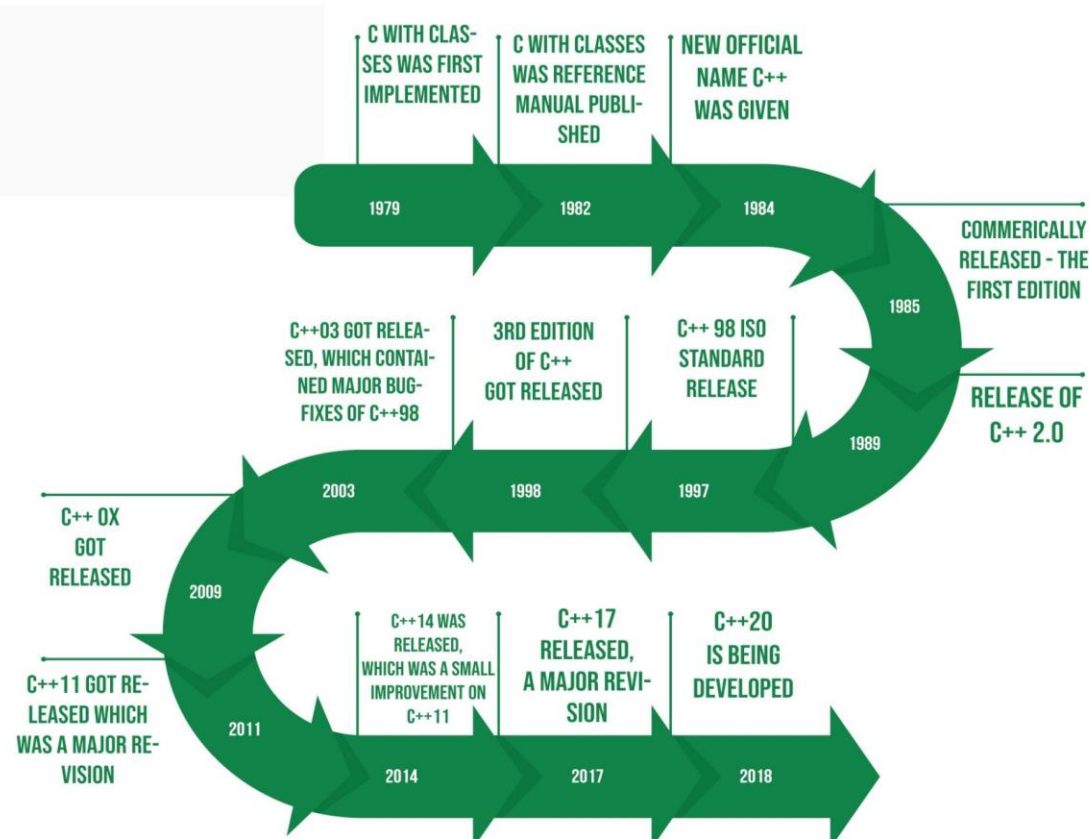


Instituto Federal do Amazonas
Disciplina: Paradigmas de Programação
Docente: Joethe Carvalho
Discentes: Carlos Souza, Rafael Kanda
Data: 16/05/2022

Defesa de Linguagem de Programação: C++

Linha do Tempo do C++

History of C++



Características Importantes

- A linguagem C++ é uma linguagem de programação orientada a objetos e é uma combinação de linguagem de baixo e alto nível – uma linguagem de nível médio.
- Essa linguagem de programação foi criada, projetada e desenvolvida por um cientista da computação dinamarquês - Bjarne Stroustrup no Bell Telephone Laboratories (agora conhecido como Nokia Bell Labs) em Murray Hill, Nova Jersey.
- C++ era inicialmente conhecido como “C com classes” e foi renomeado C++ em 1983. ‘++’ é uma abreviação para adicionar um à variedade na programação; portanto, C++ significa aproximadamente que “um maior que C”.

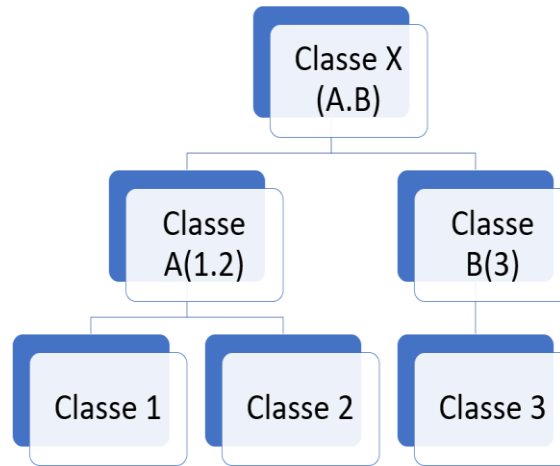


BELL TELEPHONE
LABORATORIES



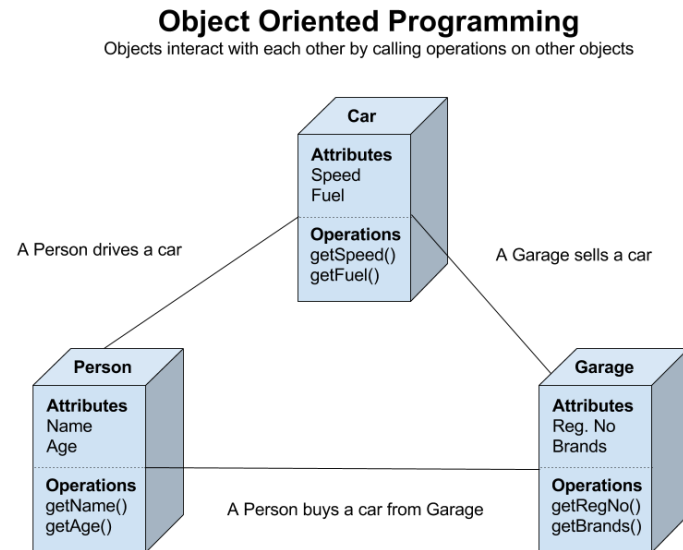
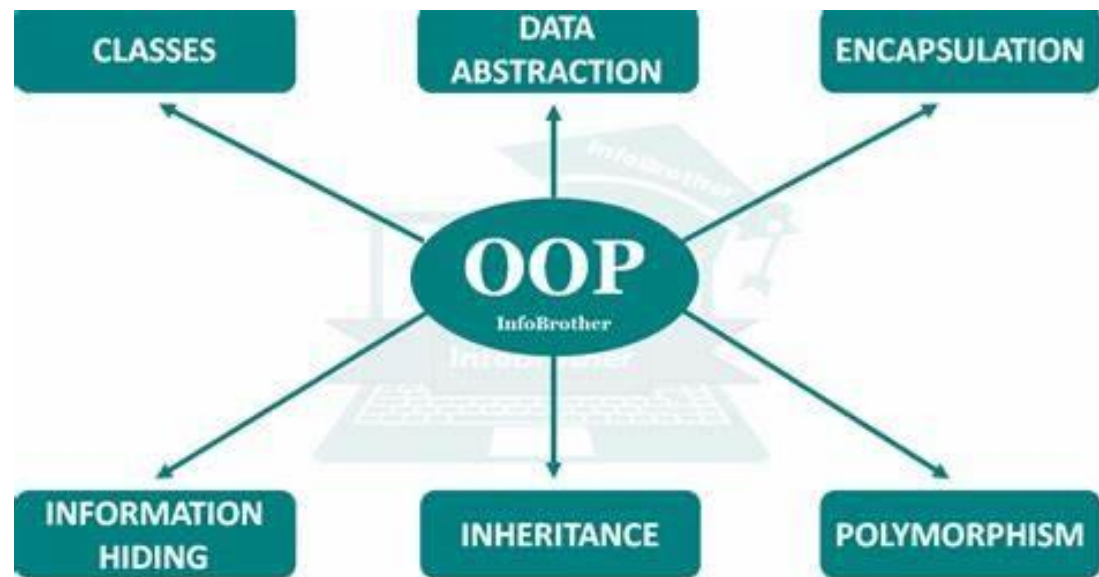
Características Peculiares do C++

- É a linguagem mais usada principalmente na programação competitiva no que diz respeito aos fatos. A maioria dos codificadores extremamente cotados normalmente usa C++ para criptografia. Você o verá em qualquer site online.
- Múltiplas heranças: Linguagens volumosas não têm essa facilidade.
- Capacidade de modularizar código, encapsulamento, polimorfismo, etc.



Por quê o C++ foi criado ?

- O principal sonho de Bjarne era colocar a Programação Orientada a Objetos, criada por ele, na linguagem C, que durante sua época ainda era a linguagem de programação amplamente respeitada por sua mobilidade, portabilidade e compacidade, não sacrificando a velocidade ou a praticidade de baixo nível (e ainda é assim até hoje).
- Sua linguagem de programação incluía inlining, herança básica, argumentos de função padrão, categorias e classificação confiável estava verificando adicionalmente todas ou quaisquer opções da linguagem C.



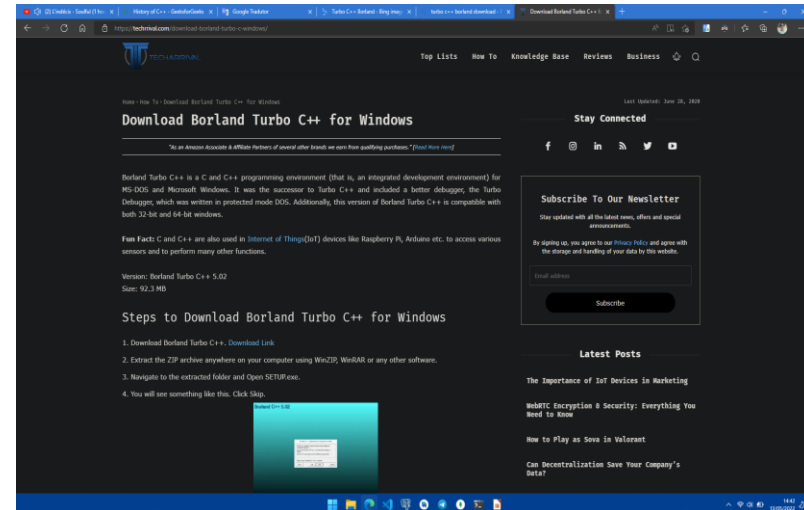
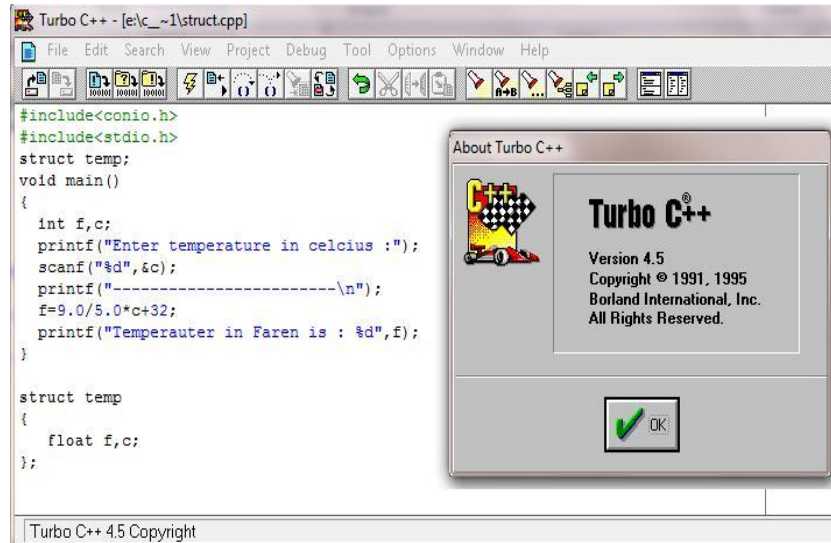
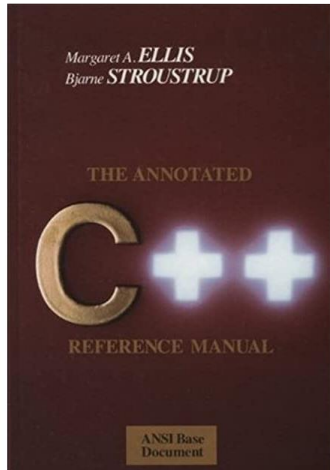
Melhores aplicações do C++

- Jogos
- Interface Gráfica do Usuário
- Software de Banco de Dados
- Sistema Operacional
- Browsers
- Computação Avançada e Gráficos
- Programas de Banco
- Sistemas de Nuvem
- Compiladores
- Sistemas Embarcados
- Sistema de Empresas
- Bibliotecas



O Sucesso do Turbo C++

- Em 1990, o manual “The Annotated C++” foi lançado por toda parte. Somente em 1990, no mesmo e idêntico ano, o compilador Turbo C++ da Borland também foi lançado comercialmente como um produto publicitário.
- O Turbo C++ adiciona um excesso de outras bibliotecas que podem ter um impacto substancial no desenvolvimento do C++. E embora a última atualização estável do Turbo C++ tenha sido em 2006, o compilador continua sendo amplamente utilizado.



Exemplos de Códigos em C++

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

class FunctionOverloading
{
public:
    void calculation(int a, int b, int k) /*addition of two numbers*/
    {
        cout << "The sum is = " << a + b + k << endl;
    }

    void calculation(double c, double d, double e) /*Multiplication of two
numbers*/
    {
        cout << "Multiplication is = " << c * d * e << endl;
    }

    int calculation(int f, int g)
    {
        return (f / g);
    }
};

int main()
{
    FunctionOverloading obj1;

    obj1.calculation(5, 6, 7);
    obj1.calculation(2.34, 4.54, 6.72);
    cout << "The division is=" << obj1.calculation(10, 5) << endl;
}
```

```
#include<iostream>
using namespace std;
class CodesCracker
{
    private:
        float r;
    public:
        void getData();
        float findArea();
        float findCircum();
};

void CodesCracker::getData()
{
    cout<<"Enter the Radius of Circle: ";
    cin>>r;
}

float CodesCracker::findArea()
{
    return (3.14*r*r);
}

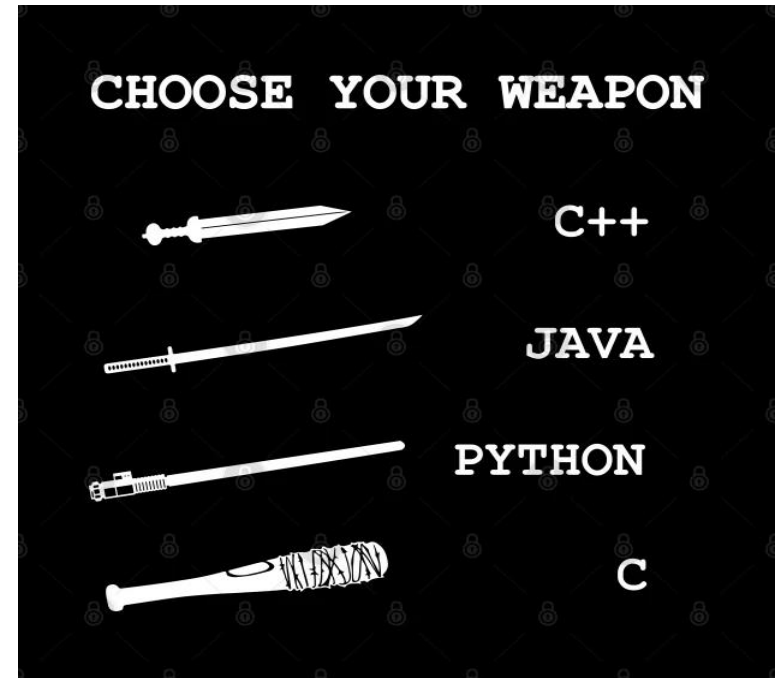
float CodesCracker::findCircum()
{
    return (2*3.14*r);
}

int main()
{
    CodesCracker c;
    c.getData();
    cout<<"\nArea of Circle = "<<c.findArea();
    cout<<"\nCircumference of Circle = "<<c.findCircum();
    cout<<endl;
    return 0;
}
```


Conclusão

- O que Bjarne Stroustrup fez foi “adicionar P.O.O na Language C”, tratando o código como objetos vivos. O que torna o C++ tão bom é que possui a velocidade do C, e também é uma linguagem de programação de alto nível, portanto, nos permite dizer que possui o melhor de cada mundo.
- No entanto, por outro lado, C++ é difícil de ser notado para iniciantes e, para um código equivalente que você escreveu em C++, você poderá escrever em Python com 25% do tempo gasto.

| Python vs C++ | |
|---|---|
| #1 | |
| <u>Advantages :</u> <ul style="list-style-type: none">- Easy to learn- Easy to access libraries- Scientific community sharing (open source, many libraries) <u>Disadvantages:</u> <ul style="list-style-type: none">- Slow- Interpreted (dependencies)- Not typed (errors at runtime) | <u>Advantages :</u> <ul style="list-style-type: none">- Execution speed- Pre-Compiled (exe on machine)- Typed (well defined)- Modern professional libraries <u>Disadvantages:</u> <ul style="list-style-type: none">- Learning curve- Harder to acces libraries (less sharing than in Python) |



Fontes

- <https://www.geeksforgeeks.org/history-of-c>
- <https://www.softwaretestinghelp.com/cpp-applications/>
- <https://codescracker.com/cpp/program/cpp-program-calculate-area-circumference.htm>
- <https://morioh.com/p/2f18dab77462>