

# Projeto Final

Algoritmos e Técnicas de  
Programação

**Nome:** Jonathan Nicolas Pipo de Lima

## Introdução

**Tema:** Análise das taxas anuais de crescimento real do PIB (%) entre Brasil, Estados Unidos e China.

É inegável que o Brasil sofre com problemas econômicos, seja pela burocracia, corrupção ou aplicação de políticas ultrapassadas. Logo, a análise da taxa de crescimento real do PIB pode estimar numericamente o grau de avanço ou de retrocesso da economia.

## Objetivo

O programa tem como objetivo analisar uma base de dados relativos aos PIB's de três países e ajudar o usuário a identificar aspectos estatísticos referente a economia dos países estudados.

## Metodologia

- Levantamento dos dados:

A escolha dos países que seriam analisados foi feita pensando num modo que os dados fizessem algum sentido. O Brasil, logicamente, é o país principal a ser analisado. Ademais, os outros dois foram escolhidos como exemplos de comparação. São eles: Estados Unidos e China, dois países exemplares no que tange a economia mesmo possuindo políticas divergentes.

- Os dados:

Os dados são referentes aos últimos 30 anos dos países analisados, a partir de 2019.

Produto Interno Bruto (PIB) - Taxa de Crescimento Real (%) x Ano

	Brasil	EUA	China
2019	-4,3	-2,6	4
2018	-3,8	-0,1	4,2
2017	-3,5	0,3	6,7
2016	-0,5	1,1	6,8
2015	-0,2	1,6	6,9
2014	-0,2	1,6	6,9
2013	0,4	1,7	6,9
2012	0,5	1,9	7,3
2011	0,5	2	7,4

2010	0,9	2,2	7,6
2009	1	2,2	7,7
2008	1	2,2	7,8
2007	1	2,4	7,9
2006	1	2,4	8
2005	1,1	2,5	8
2004	1,9	2,7	8,3
2003	2,2	2,8	8,5
2002	2,3	2,8	9
2001	2,3	2,9	9,1
2000	2,7	3,1	9,1
1999	3,2	3,2	9,1
1998	3,3	3,2	9,2
1997	3,7	3,5	9,2
1996	4,2	3,7	10,1
1995	4,2	3,8	10,2
1994	4,5	4	10,3
1993	5,1	4,4	11,5
1992	5,1	4,4	11,9
1991	5,4	4,5	13,7
1990	5,5	4,8	14,1
1989	7,5	5	14,3

- **A implementação:**  
O programa foi desenvolvido em C++.
- **Avaliação de resultados:**  
Para avaliar os resultados obtidos do programa foi utilizado o Excel, com o intuito de validar as operações matemáticas feitas pelo programa.

## Descrição do Código

- Os vetores utilizados foram:

```
int pano[31] = {2019, 2018, 2017, 2016, 2015, 2014, 2013, 2012, 2011, 2010,
2009, 2008, 2007, 2006, 2005, 2004, 2003, 2002, 2001, 2000, 1999, 1998,
1997, 1996, 1995, 1994, 1993, 1992, 1991, 1990, 1989};
```

“Int pano” são os anos analisados armazenados em um vetor de 31 índices.

```
double pbrasil[31] = {1, 1, 1.1, -3.5, -3.8, 0.5, 2.3, 0.9, 2.7, 7.5, -0.2, 5.1, 5.4,
3.7, 2.3, 5.1, -0.2, 1, 1.9, 4.2, 0.5, 0.4, 3.3, 2.2, 4.2, 5.5, 4.5, -0.5, 1, -4.3, 3.2};
```

“Double pbrasil” são os pibs do Brasil dos últimos 31 anos armazenados num vetor de 31 índices.

double peua[31] = {2.2, 2.9, 2.2, 1.6, 2.4, 2.5, 1.6, 2.2, 1.7, 2.8, -2.6, 1.1, 2, 3.2, 3.2, 4.4, 3.1, 2.4, 0.3, 5, 4.8, 4.5, 4.4, 3.8, 2.7, 4, 2.8, 3.5, -0.1, 1.9, 3.7};

“Double peua” são os PIBS dos EUS dos últimos 31 anos armazenados num vetor de 31 índices.

double pchina[31] = {6.9, 6.9, 6.8, 6.7, 6.9, 7.3, 7.7, 7.8, 9.2, 10.3, 9.1, 9, 11.9, 10.2, 9.1, 9.1, 8, 8, 8.3, 7.4, 7.6, 7.9, 9.2, 10.1, 11.5, 13.7, 14.1, 14.3, 8.5, 4, 4.2};

“Double pchina” são os PIBS da China dos últimos 31 anos armazenados num vetor de 31 índices.

float ordfreqvalorbr[23] = {-4.3, -3.8, -3.5, -0.5, -0.2, 0.4, 0.5, 0.9, 1, 1.1, 1.9, 2.2, 2.3, 2.7, 3.2, 3.3, 3.7, 4.2, 4.5, 5.1, 5.4, 5.5, 7.5};

“float ordfreqvalorbr” são os valores utilizados para calcular e ordenar as frequências.

float ordfreqfibr[23] = {1, 1, 1, 1, 2, 1, 2, 1, 4, 1, 1, 1, 2, 1, 1, 1, 1, 2, 1, 2, 1, 1, 1};

“float ordfreqfibr” são os valores de frequência absoluta.

float ordfreqffibr[23] = {1, 2, 3, 4, 6, 7, 9, 10, 14, 15, 16, 17, 19, 20, 21, 22, 23, 25, 26, 28, 29, 30, 31};

“float ordfreqffibr” são os valores de frequência acumulada.

float ordfreqfrbr[23] = {3.22, 3.22, 3.22, 3.22, 6.45, 3.22, 12.90, 3.22, 3.22, 3.22, 6.45, 3.22, 3.22, 3.22, 6.45, 3.22, 6.45, 3.22, 3.22, 3.22};

“float ordfreqfrbr” são os valores de frequência relativa.

float ordfreqfibr[23] = {3.22, 6.44, 9.66, 12.88, 19.33, 22.55, 29, 32.22, 45.12, 48.34, 51.56, 54.78, 61.23, 64.45, 67.67, 70.89, 74.11, 80.56, 83.78, 90.23, 93.45, 96.67, 99.89};

“float ordfreqfibr” são os valores de frequência relativa acumulada.

float ordfreqvaloreua[24] = {-2.6, -0.1, 0.3, 1.1, 1.6, 1.7, 1.9, 2, 2.2, 2.4, 2.5, 2.7, 2.8, 2.9, 3.1, 3.2, 3.5, 3.7, 3.8, 4, 4.4, 4.5, 4.8, 5};

“float ordfreqvaloreua” são os valores utilizados para calcular e ordenar as frequências.

float ordfreqfieua[24] = {1, 1, 1, 1, 2, 1, 1, 1, 3, 2, 1, 1, 2, 1, 1, 2, 1, 1, 1, 1, 2, 1, 1, 1};

“float ordfreqfieua” são os valores de frequência absoluta.

`float ordfreqffieua[24] = {1, 2, 3, 4, 6, 7, 8, 9, 12, 14, 15, 16, 18, 19, 20, 22, 23, 24, 25, 26, 28, 29, 30, 31};`

“float ordfreqffieua” são os valores de frequência acumulada.

`float ordfreqfreua[24] = {3.22, 3.22, 3.22, 3.22, 6.45, 3.22, 3.22, 3.22, 9.67, 6.45, 3.22, 3.22, 6.45, 3.22, 3.22, 6.45, 3.22, 3.22, 3.22, 6.45, 3.22, 3.22, 3.22};`

“float ordfreqfreua” são os valores de frequência relativa.

`float ordfreqfrieua[24] = {3.22, 6.44, 9.66, 12.88, 19.33, 22.55, 25.77, 28.99, 28.66, 45.11, 48.33, 51.55, 58, 61.22, 64.44, 70.89, 74.11, 77.33, 80.55, 83.77, 90.22, 93.44, 96.66, 99.88};`

“float ordfreqfrieua” são os valores de frequência relativa acumulada.

`float ordfreqvalorchina[25] = {4, 4.2, 6.7, 6.8, 6.9, 7.3, 7.4, 7.6, 7.7, 7.8, 7.9, 8, 8.3, 8.5, 9, 9.1, 9.2, 10.1, 10.2, 10.3, 11.5, 11.9, 13.7, 14.1, 14.3};`

“float ordfreqvalorchina” são os valores utilizados para calcular e ordenar as frequências.

`float ordfreqfichina[25] = {1, 1, 1, 1, 3, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 3, 2, 1, 1, 1, 1, 1, 1, 1, 1};`

“float ordfreqfichina” são os valores de frequência absoluta.

`float ordfreqffichina[25] = {1, 2, 3, 4, 7, 8, 9, 10, 11, 12, 13, 15, 16, 17, 18, 21, 23, 24, 25, 26, 27, 28, 29, 30, 31};`

“float ordfreqffichina” são os valores de frequência acumulada.

`float ordfreqfrchina[25] = {3.22, 3.22, 3.22, 3.22, 9.67, 3.22, 3.22, 3.22, 3.22, 3.22, 3.22, 6.45, 3.22, 3.22, 3.22, 9.67, 6.45, 3.22, 3.22, 3.22, 3.22, 3.22, 3.22, 3.22, 3.22};`

“float ordfreqfrchina” são os valores de frequência relativa.

`float ordfreqfrichina[25] = {3.22, 6.44, 9.66, 12.88, 22.55, 25.77, 28.99, 32.21, 35.43, 38.65, 41.87, 45.09, 51.54, 54.76, 57.98, 61.20, 70.87, 77.32, 80.54, 83.76, 86.98, 90.20, 93.42, 96.64, 99.86};`

“float ordfreqfrichina” são os valores de frequência relativa acumulada.

- As funções utilizadas foram:

```
float media(float pais[], int x){  
    float media, tot;  
    for(int i = 0; i<x; i++){  
        tot += pais[i];  
    }  
    media = tot/x;  
    return media;  
}
```

Utilizada para calcular a média mais rapidamente.

```
void ordenar(float vetor[], int tamanho){  
    float y, x, aux;  
  
    for(int x = 0; x < 31; x++){  
        for(int y = x; y< 31; y++){  
            if (vetor[x] > vetor[y]){  
                aux = vetor[x];  
                vetor[x] = vetor[y];  
                vetor[y] = aux;  
            }  
        }  
    }  
}
```

Utilizada para ordenar os valores dos vetores mais rapidamente.

```
void mediana(float pais[], int med ){  
    med = med / 2;  
    ordenar(pais, 30);  
    cout << "A mediana é: " << pais[med];  
}
```

Utilizada para calcular a mediana mais rapidamente.

```
float variancia(float s[], int n){
    float sum = 0.0;
    float dev = 0.0;
    float med = media(s , n);
    int i = 0;

    for( i = 0; i < n; i++ )
    {
        dev = s[i] - med;
        sum += (dev * dev);
    }

    cout << sum / 31;
}
```

Utilizada para calcular a variância mais rapidamente.

```
void moda(float v[]){
    int T = 32; //tamanho do vetor
    int i, j, cont[T];
    float conta, moda;

    for(i=0;i<T;i++){
        for(j=i+1;j<T;j++){
            if(v[i]==v[j]){
                cont[i]++;
                if(cont[i]>conta){
                    conta=cont[i];
                    moda=v[i];
                }
            }
        }
    }
}
```

```

        }
    }
    cont[i]=0;
}
if(conta == 0){
    printf("Nao existe moda\n");
}
else{
    printf("Moda: %.2f\n", moda);
}
}

```

Utilizada para calcular a moda mais rapidamente.

```

void minima(float pais[]){
    float menor = 10;
    for(int i = 0; i<31; i++){
        if(pais[i]<menor){
            menor = pais[i];
        }
    }
    cout << "O menor valor é: " << menor ;
}

```

Exibir o menor valor rapidamente.

```

void maxima(float pais[]){
    float maior;
    for(int i = 0; i<31; i++){
        if(pais[i]>maior){
            maior = pais[i];
        }
    }
}

```



```

        cout << "O maior valor é: " << maior;
    }

```

Exibir o maior valor rapidamente.

O programa inicia dando ao usuário cinco alternativas:

1 >. Exibir o país que possui a maior taxa média de crescimento dos últimos trinta anos.

2 >. Exibir a taxa de crescimentos dos últimos 'n' anos do país solicitado.

3 >. Exibir o período de maior crescimento de determinado país.

4>. Exibir menor valor, maior valor, média, moda e variância de cada país.

4 >. Exibir as ordenações de frequência dos três países.

Ao selecionar a primeira opção o programa identifica no banco de dados qual dos países possui a maior taxa de crescimento dos últimos trinta anos.

```

    double resbrasil, reseua, reschina, resbra;
    reschina = 0;
    resbrasil = 0;
    reseua = 0;
    for(int i = 0; i<31; i++){
        resbrasil = resbrasil + pbrasil[i];
        mbrasil = resbrasil / 31;
    }
    for(int i = 0; i<31; i++){
        reseua = reseua + peua[i];
        meua = reseua / 31;
    }
    for(int i = 0; i<31; i++){
        reschina = reschina + pchina[i];
        mchina = reschina / 31;
    }

```

Em seguida o programa mostra ao usuário qual país possui a maior taxa de crescimento dos últimos 30 anos.

```

    if(mbrasil > mchina && mbrasil > meua){
        cout << "O país com a maior taxa de crescimento dos
últimos 30 anos é o Brasil com " << mbrasil << "% de taxa de crescimento.";
    }

```

```

    }
    else if (mchina > mbrasil && mchina > mea) {
        cout << "O país com a maior taxa de crescimento dos
últimos 30 anos é a China com " << mchina << "% de taxa de crescimento.";
    }
    else {
        cout << "O país com maior a taxa de crescimento dos
últimos 30 anos são os Estados Unidos com " << mea << "% de taxa de
crescimento.";
    }
}

```

Ao seleccionar a opção 2 o programa pede para o usuário digitar o país que será analisado.

```

cout << "Digite o país a ser analisado. Brasil, China, EUA\n";

scanf("%s", &pais);

```

Em seguida, o usuário deve digitar o país que quer analisar e o número de anos que serão analisados.

```

if (!strcmp(&pais, "Brasil") || !strcmp(&pais, "brasil")){
    cout << "Digite o número de anos a ser
analisado.\n";

    scanf("%i", &ano);
    for (int j = 0; j<ano; j++){
        cout << "A taxa de variação do " << &pais <<
" é de " << pbrasil[j] << "% referente ao ano de " << pano[j] << "\n";
    }
}
else if (!strcmp(&pais, "China") || !strcmp(&pais, "china")){
    cout << "Digite o número de anos a ser
analisado.\n";

    scanf("%i", &ano);
    for (int j = 0; j<ano; j++){
        cout << "A taxa de variação da " << &pais <<
" é de " << pchina[j] << "% referente ao ano de " << pano[j] << "\n";
    }
}
}

```

```

else if (!strcmp(&pais, "EUA") || !strcmp(&pais, "eua")){
    cout << "Digite o número de anos a ser
    analisado.\n";

    scanf("%i", &ano);
    for (int j = 0; j<ano; j++){
        cout << "A taxa de variação dos " << &pais<<
        " é de " << peua[j] << "% referente ao ano de " << pano[j] << "\n";
    }
}

```

Caso o programa não reconheça o que foi digitado.

```

else {
    cout << "Nenhum país encontrado, digite Brasil,
    China ou EUA.";
}

```

Caso o usuário selecione a opção 3, o programa pede para o usuário digitar o país a ser analisado. Após isso, o programa mostra ao usuário a maior taxa de crescimento e o ano referente a essa taxa.

```

double maior;

    int anoesp; //Refere-se ao ano específico da maior
    alteração da taxa do PIB

    maior = 0;

    cout << "Digite o país a ser analisado. Brasil, China,
    EUA\n";

    scanf("%s", &pais);
    if (!strcmp(&pais, "Brasil") || !strcmp(&pais, "brasil")){

        for(int i = 0; i<31; i++){
            if(pbrasil[i]>maior){
                maior = pbrasil[i];
                anoesp = i;
            }
        }

        cout << maior;
        cout << " referente ao ano de " << pano[anoesp];
    }
}

```

```

    }
    else if (!strcmp(&pais, "China") || !strcmp(&pais, "china")){
        for(int i = 0; i<31; i++){
            if(pchina[i]>maior){
                maior = pchina[i];
                anoesp = i;
            }
        }
        cout << maior;
        cout << " referente ao ano de " << pano[anoesp];
    }
    else if (!strcmp(&pais, "EUA") || !strcmp(&pais, "eua")){
        for(int i = 0; i<31; i++){
            if(peua[i]>maior){
                maior = peua[i];
                anoesp = i;
            }
        }
        cout << maior;
        cout << " referente ao ano de " << pano[anoesp];
    }
    else {
        cout << "Nenhum país encontrado, digite Brasil,
China ou EUA.";
    }
}

```

Caso o usuário selecione a opção 4, o programa irá exibir o menor valor, o maior valor, a média, a mediana, a moda e a variância dos três países.

```

cout << "\n";

cout << "----- Os dados do BRASIL são: ----- \n";
minima(pbrasil);
cout << "\n";
maxima(pbrasil);

```

```

cout << "\n";
cout << "A média é: " << media(pbrasil, 31);
cout << "\n";
mediana(pbrasil, 31);
cout << "\n";
moda(pbrasil);
cout << "A variancia é: ";
variancia(pbrasil, 31);
cout << "\n";

cout << "\n";
cout << "----- Os dados dos ESTADOS UNIDOS são: --
-----\n";

minima(peua);
cout << "\n";
maxima(peua);
cout << "\n";
cout << "A média é: " << media(peua, 31);
cout << "\n";
mediana(peua, 31);
cout << "\n";
moda(peua);
cout << "A variancia é: ";
variancia(peua, 31);
cout << "\n";

cout << "\n";
cout << "----- Os dados da CHINA são: ----- \n";
minima(pchina);
cout << "\n";
maxima(pchina);
cout << "\n";

```

```

cout << "A média é: " << media(pchina, 31);
cout << "\n";
mediana(pchina, 31);
cout << "\n";
moda(pchina);
cout << "A variancia é: ";
variancia(pchina, 31);
cout << "\n";

```

Caso o usuário selecione a quinta opção, o programa irá exibir as ordenações de frequência de cada país

//Brasil

```

cout << "\nOrdenação de frequência do Brasil.\n";
cout << "Valor    |";
cout << "fi    |" ;
cout << "Fi    |";
cout << "fr    |";
cout << "Fri    \n";
for (int i=0; i<23; i++){

    printf ("%0.1f", ordfreqvalorbr[i]);
    cout << "    ";
    printf ("%0.f", ordfreqfibr[i]);
    cout << "    ";
    printf ("%0.f", ordfreqffibr[i]);
    cout << "    ";
    printf ("%0.2f", ordfreqfrbr[i]); cout << "%";
    cout << "    ";
    printf ("%0.2f", ordfreqfibr[i]); cout << "%";
}

```

```

        cout << " ";
        cout << "\n";

    }
    cout << "\nOrdenação de frequência dos Estados
Unidos.";

    cout << "\n";
    cout << "Valor    |";
    cout << "fi    |" ;
    cout << "Fi    |";
    cout << "fr    |";
    cout << "Fri    \n";

    for (int j=0; j<24; j++){

        printf ("%0.1F", ordfreqvaloreua[j]);
        cout << "    ";
        printf ("%0.F", ordfreqfieua[j]);
        cout << "    ";
        printf ("%0.F", ordfreqffieua[j]);
        cout << "    ";
        printf ("%0.2F", ordfreqfreua[j]); cout << "%";
        cout << "    ";
        printf ("%0.2F", ordfreqfrieua[j]); cout << "%";

        cout << " ";
        cout << "\n";

    }

//China
    cout << "\nOrdenação de frequência da China.\n";
    cout << "Valor    |";
    cout << "fi    |" ;

```

```

cout << "Fi   |";
cout << "fr   |";
cout << "Fri   \n";
    for (int k=0; k<25; k++){

        printf ("%0.1f", ordfreqvalorchina[k]);
        cout << "   ";
        printf ("%0.F", ordfreqfichina[k]);
        cout << "   ";
        printf ("%0.F", ordfreqffichina[k]);
        cout << "   ";
        printf ("%0.2f", ordfreqfrchina[k]); cout <<
"%";

        cout << "   ";
        printf ("%0.2f", ordfreqfrichina[k]); cout <<
"%";

        cout << " ";
        cout << "\n";

    }

```

## Conclusão

- **Resultado**

O programa conclui-se com êxito para o que foi proposto. O programa em si agrega a programação não modularizada e funções. A quarta opção do programa foi pensada utilizando as funções de menor valor, maior valor, média, mediana, moda e variância, pois os dados seriam reutilizados várias vezes e as funções facilitariam essa etapa do programa.

- **Dificuldades**

A maior dificuldade na confecção do programa foi a busca pelos dados e a opção número 5.

A busca de dados foi dificultada, pois os dados encontrados eram incompletos, alguns não apresentavam os últimos 30 anos e alguns



sites não apresentavam o mesmo valor para determinado ano. Com dificuldade os dados íntegros foram encontrados e conferidos. A opção número 5 insere valores e os exibe e os comandos utilizados limitam-se a “printf” e “coat”.

## Cronograma de Trabalho

09/07/2020 – Definição do tema e busca de dados;

13/07/2020 – Menu inicial;

20/07/2020 – Implementação de duas opções;

28/07/2020 – Implementação de mais duas opções;

03/08/2020 – Confecção do relatório.

## Bibliografia

<http://datatopics.worldbank.org/world-development-indicators/>

<https://mundorama.net/?p=19355>

<https://odsbrasil.gov.br/objetivo8/indicador811>

[https://www.scielo.br/scielo.php?pid=S0101-31572010000200002&script=sci\\_arttext](https://www.scielo.br/scielo.php?pid=S0101-31572010000200002&script=sci_arttext)

<https://pt.tradingeconomics.com/china/gdp-growth-annual>

<https://www.ceicdata.com/pt/indicator/china/real-gdp-growth>

<https://pt.countryeconomy.com/governo/pib/estados-unidos>

<https://infograficos.gazetadopovo.com.br/economia/pib-do-brasil/>

<https://www.indexmundi.com/g/g.aspx?c=br&v=66&l=pt>

<https://www.indexmundi.com/g/g.aspx?c=us&v=66&l=pt>

<https://www.indexmundi.com/g/r.aspx?t=0&v=66&l=pt>