

Guia de Implementação do TCC

Análise de ferramentas da Bioinformática aplicadas em estudos genéticos

Alunos

Rafael Franco Nascimento – Ra: 9947.

Rodrigo Santiago Claro Filho – Ra: 99113.

Orientador

Camilo César Perucci

Coorientador

Fabiano Berardo de Sousa

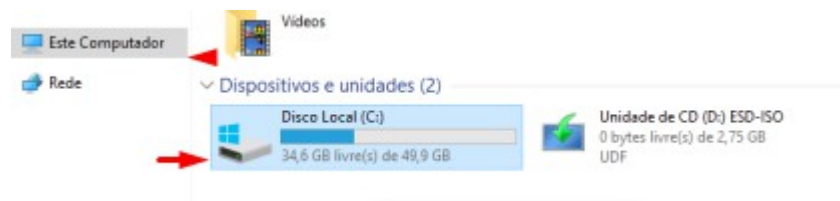
Sessão GIT (Clonando o Projeto)

Antes de começar, certifique-se que você possui o [GitBash](#) instalado em sua máquina, possuindo o mesmo, vamos aos primeiros passos!

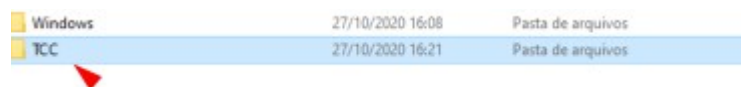
Primeiro vamos acessar o repositório do Projeto no Github: <https://github.com/DevSantiro/TCC>

Agora crie um diretório vazio no local que achar melhor, no meu caso, irei na raiz “C:/” e criar um diretório “TCC”:

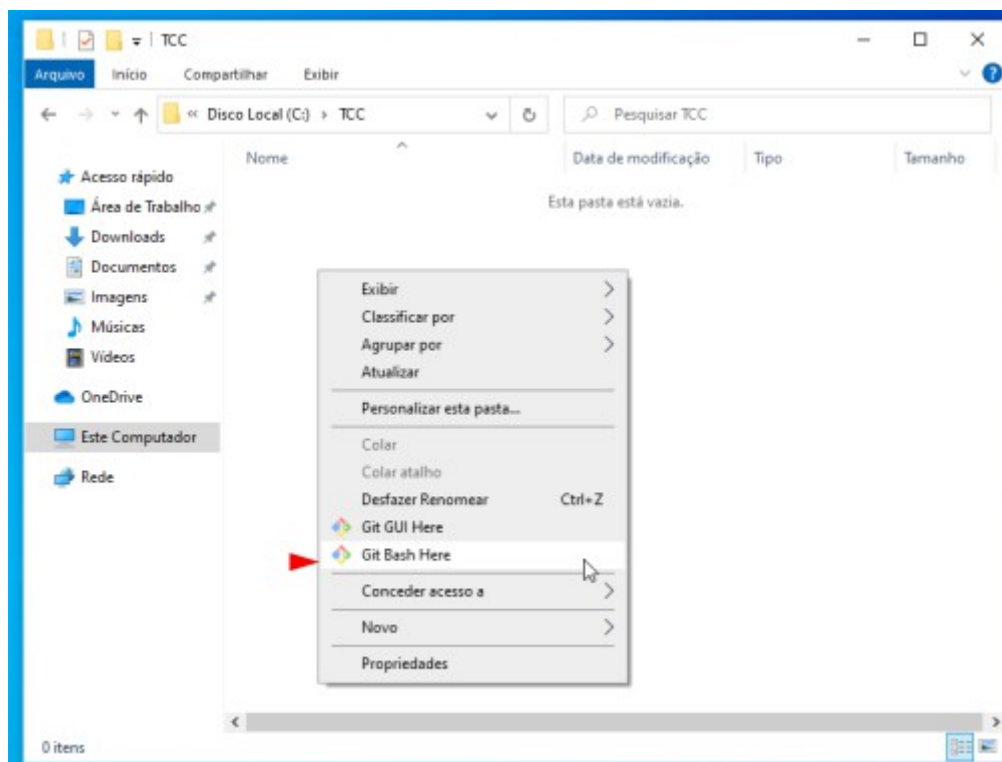
Acessando o diretório:



Criando Diretório TCC:



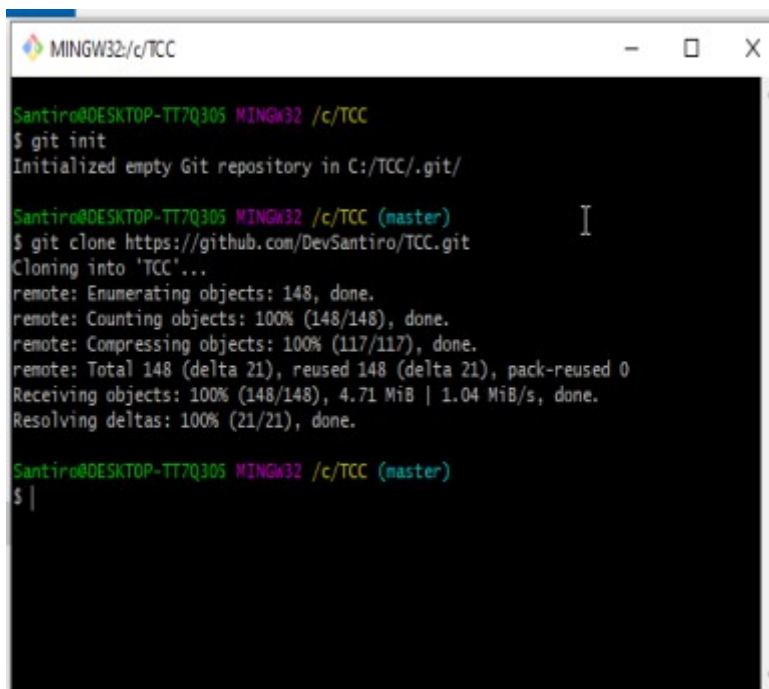
Ao criar o diretório iremos configurar o GIT para receber o Projeto, para isso precisamos entrar no diretório “TCC” clicar com o botão direito do mouse e selecionar a opção “GIT BASH HERE”:



Após realizado este procedimento será aberto um terminal do GIT, onde por linha de comandos iremos gerar um “Clone” do Projeto.

Os comandos necessários são:

- 1º “git init” todos os comandos são sem as aspas.
- 2º “git clone <https://github.com/DevSantiro/TCC.git>”



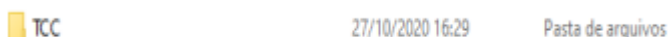
```
MINGW32/c/TCC
Santiro@DESKTOP-TT7Q305 MINGW32 /c/TCC
$ git init
Initialized empty Git repository in C:/TCC/.git/

Santiro@DESKTOP-TT7Q305 MINGW32 /c/TCC (master)
$ git clone https://github.com/DevSantiro/TCC.git
Cloning into 'TCC'...
remote: Enumerating objects: 148, done.
remote: Counting objects: 100% (148/148), done.
remote: Compressing objects: 100% (117/117), done.
remote: Total 148 (delta 21), reused 148 (delta 21), pack-reused 0
Receiving objects: 100% (148/148), 4.71 MiB | 1.04 MiB/s, done.
Resolving deltas: 100% (21/21), done.

Santiro@DESKTOP-TT7Q305 MINGW32 /c/TCC (master)
$ |
```

Dado o 2º comando, o git se encarrega de trazer todo o “esqueleto” do nosso Projeto para o diretório “TCC” em que realizamos o comando.

Ao fechar o GIT, note agora que existe um diretório chamado “TCC”:



Esse diretório é o código (esqueleto) de nosso projeto, todos os arquivos já estão disponíveis para visualização e edição caso necessário, no entanto, ainda precisamos configurar todo o ambiente, realizando o download e configuração das ferramentas (Scripts) que utilizamos junto com o Python (como o Django que é o responsável pelo nosso site na WEB – local) e dos downloads e configurações de Softwares de bioinformática.

Instalando o Python 3.8.1

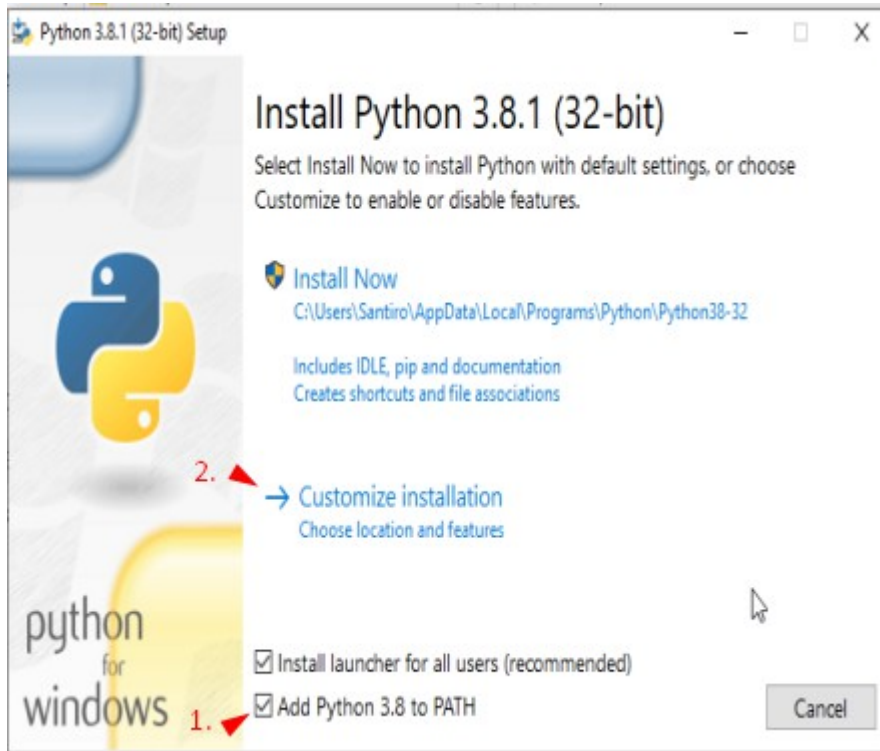
Vamos começar instalando o Python na versão em que foi desenvolvida o Projeto:

<https://www.python.org/downloads/release/python-381/>

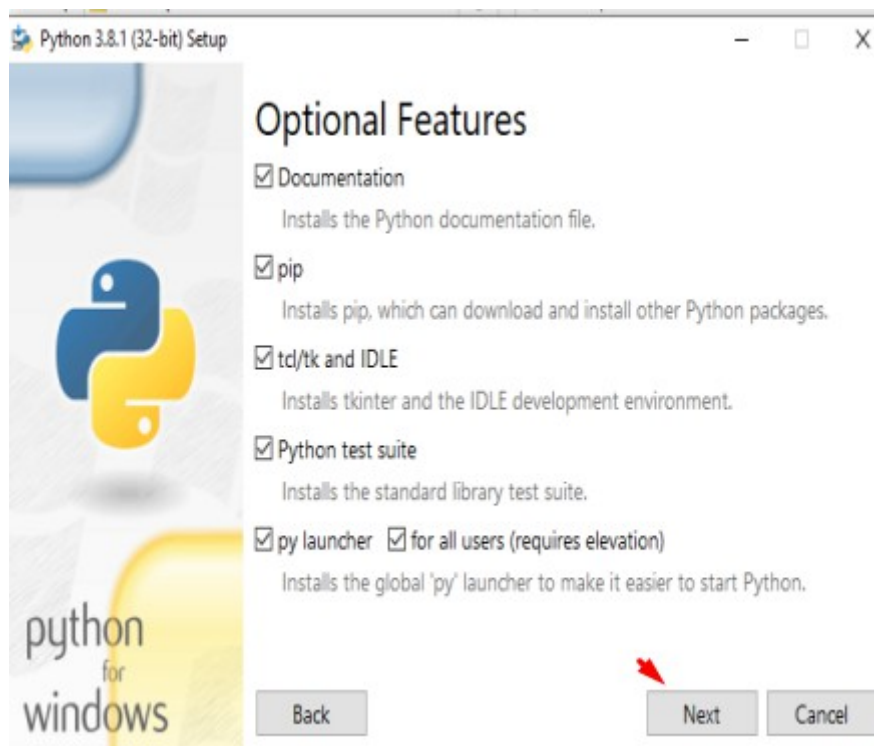
Dentro do site, navegue até a sessão “Files”, lá encontraremos todos os arquivos do Python na versão 3.8.1, como trabalhamos com Windows, escolheremos a versão do nosso sistema operacional (32 bits = Windows x86 executable installer ou 64 bits = Windows x86-64 executable installer).

No meu caso, estou instalando em uma máquina virtual com Windows 32 bits, então irei selecionar a opção: **Windows x86 executable installer** para Download.

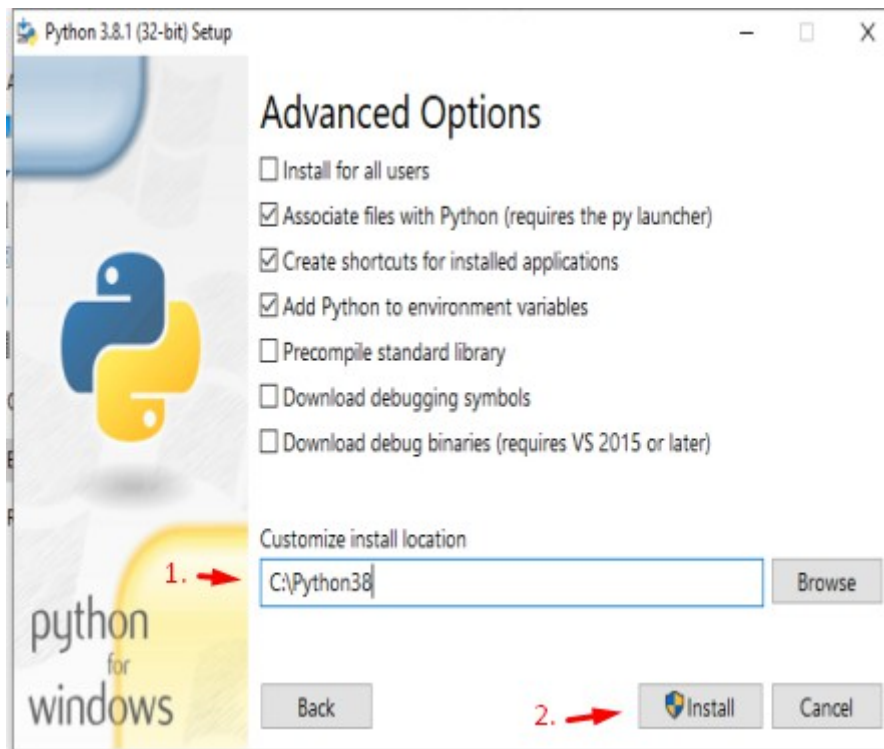
O processo de instalação é simples, ao realizar o Download, clique 2 vezes no “python-3.8.1”. Na primeira tela iremos selecionar a opção “Customize installation” para escolhermos o diretório em que instalaremos o Python, selecionando também a opção de “Add Python 3.8 to PATH”, o que facilita o nosso trabalho de configuração.



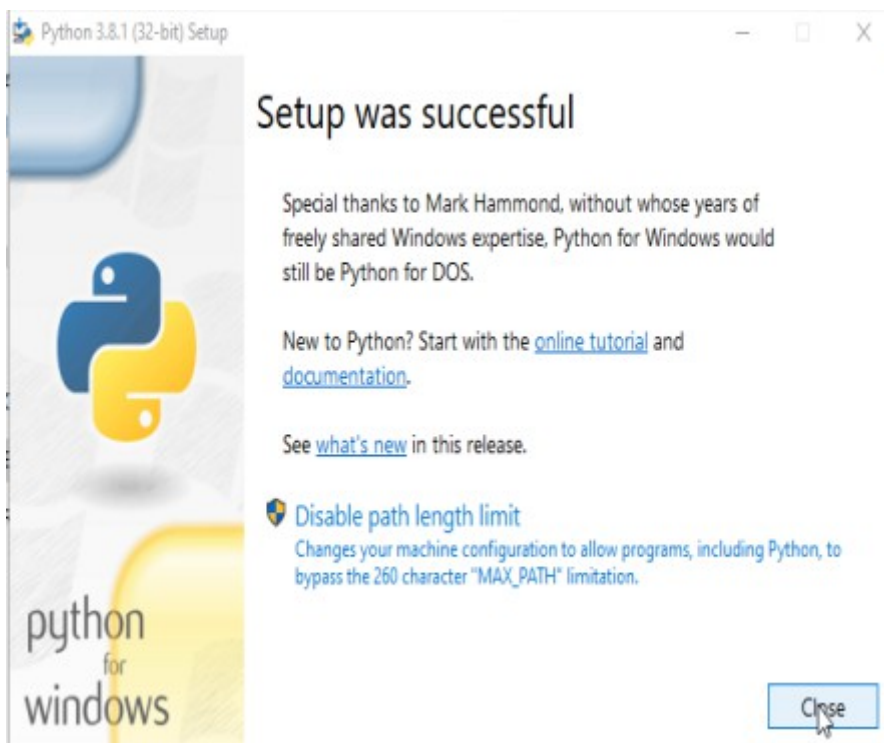
Selecione “Next” na segunda janela



Na terceira janela, mudaremos as configurações na “caixa texto” para “[C:\Python38](#)”, sem aspas.

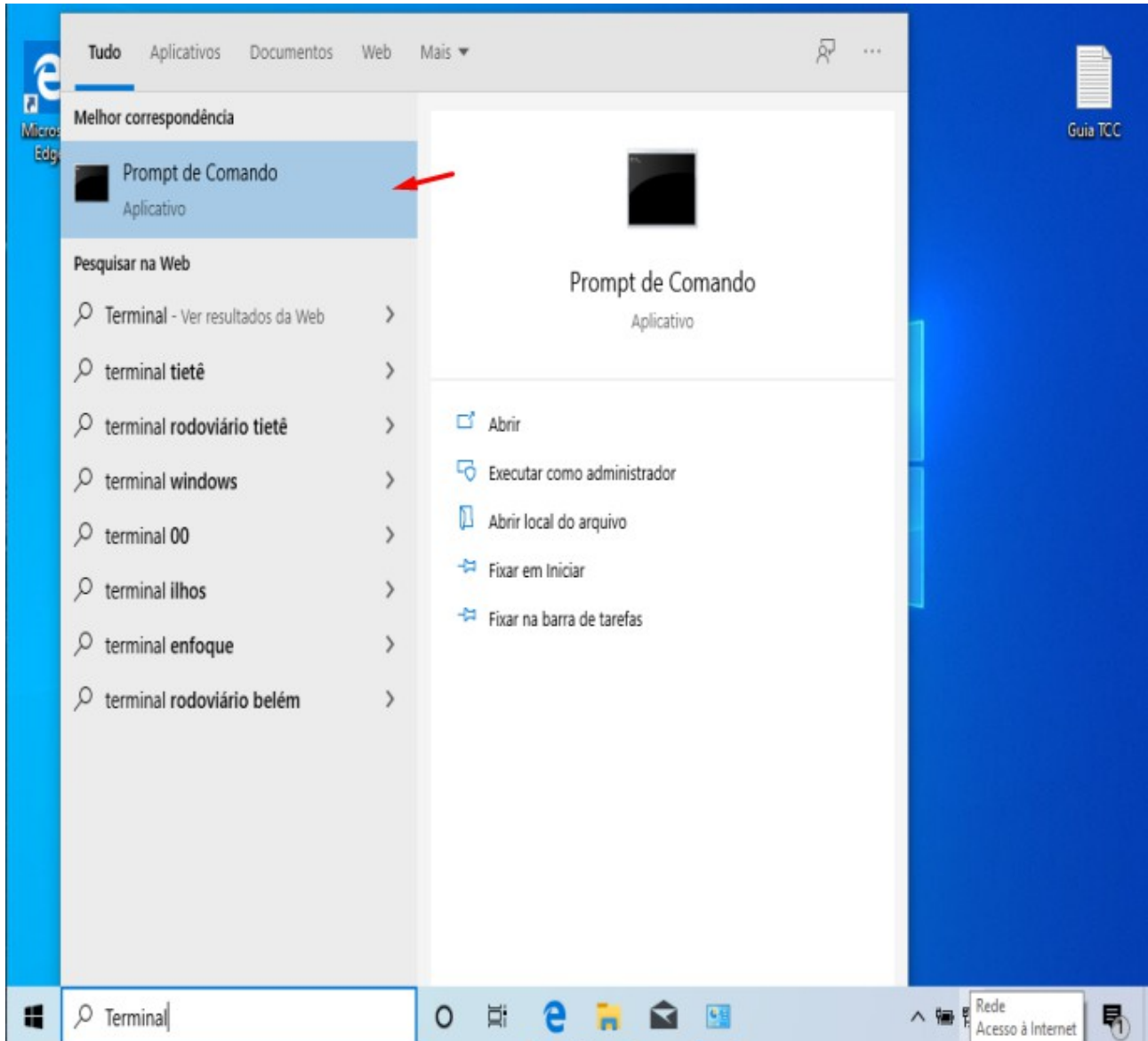


Na quarta janela clique em “Close” para finalizar o instalador.



Feita a instalação, vamos verificar se o Python realmente foi adicionado a nossa variável de ambiente!

1º Passo: Aperte a tecla do “Windows” no seu teclado e pesquise por “Terminal”, clique em “Prompt de Comando”



2º Com o “Prompt de Comando” ou simplesmente “Terminal” aberto, escreva “python” e de “Enter”

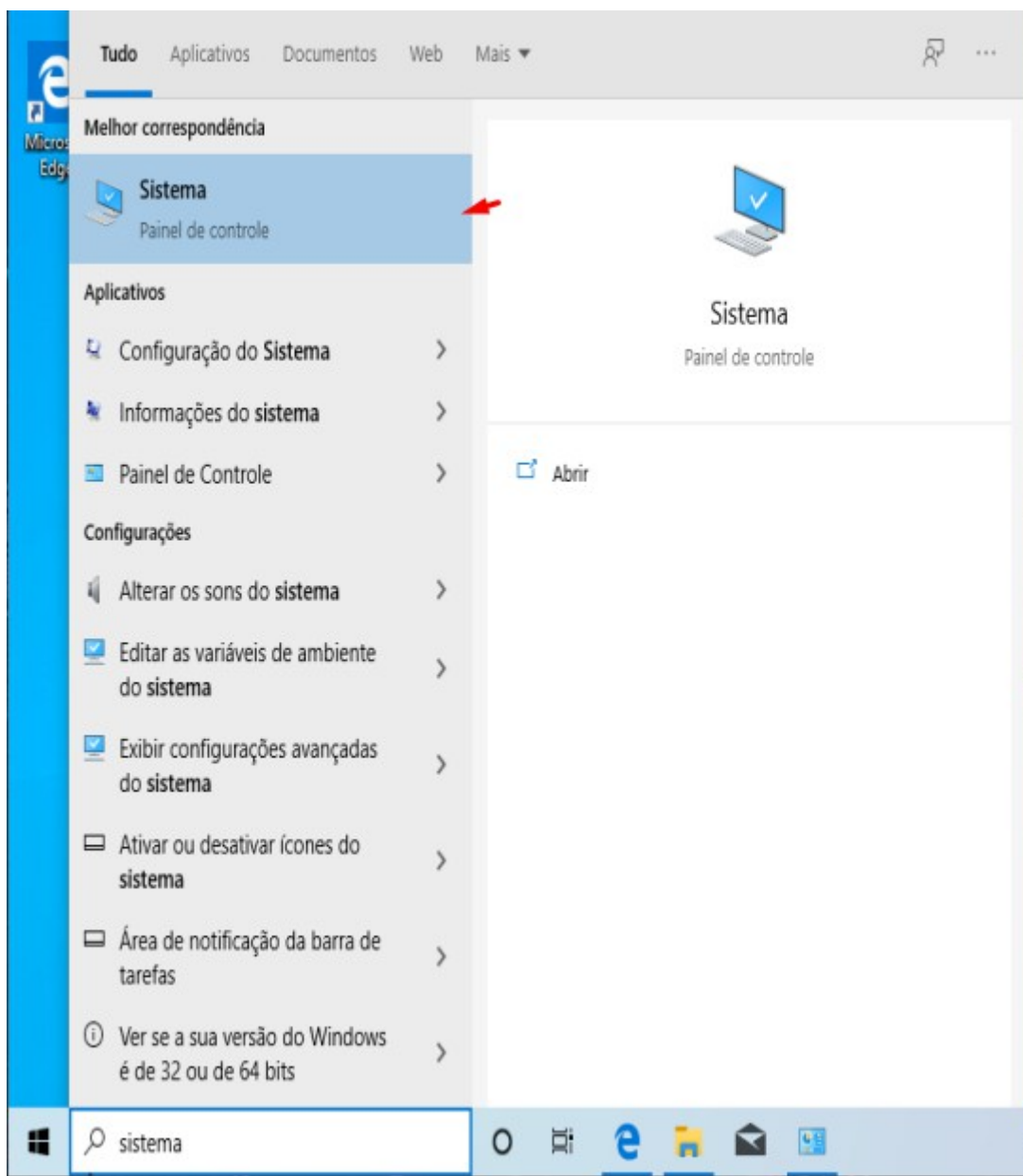
```
C:\Windows\system32\cmd.exe - python
Microsoft Windows [versão 10.0.19041.450]
(c) 2020 Microsoft Corporation. Todos os direitos reservados.
C:\Users\Santiro>python
```

Caso você obtenha essas linhas como resposta, o seu Python foi instalado e adicionado ao PATH corretamente, pode seguir para o próximo tópico!

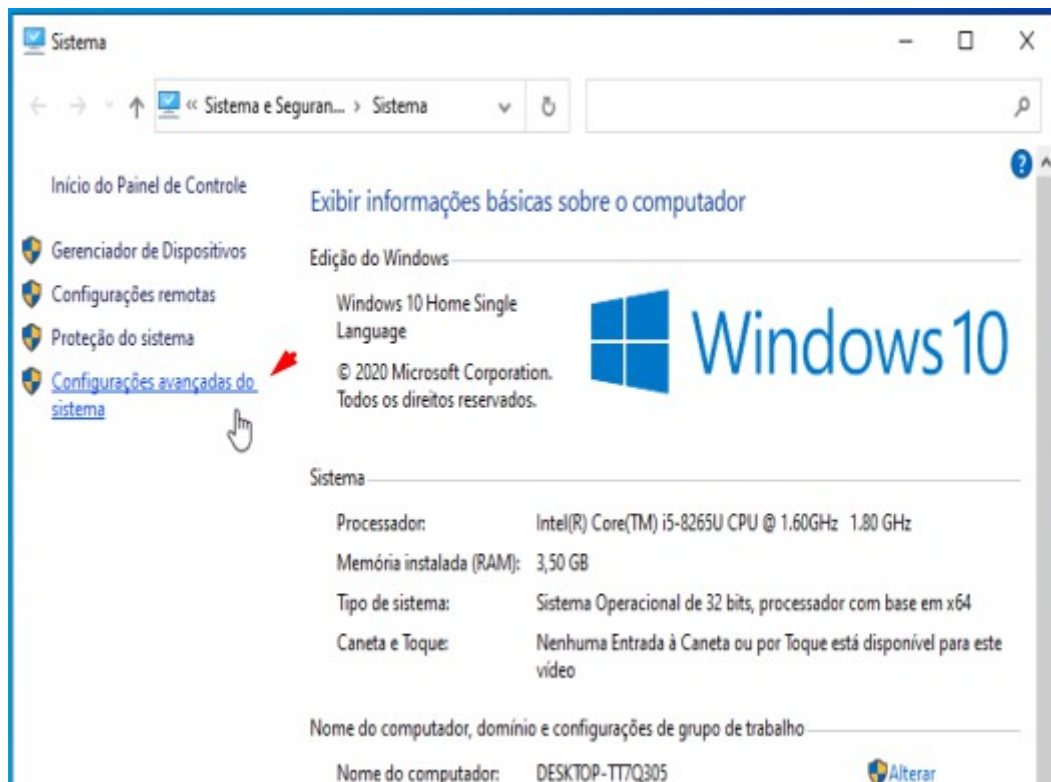
```
Python 3.8.1 (tags/v3.8.1:1b293b6, Dec 18 2019, 22:39:24) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
```

No entanto caso você tenha obtido algum erro como “python não é reconhecido” provavelmente o processo de instalação não adicionou o “.exe” do Python ao PATH do Windows. Então vamos adicionar manualmente.

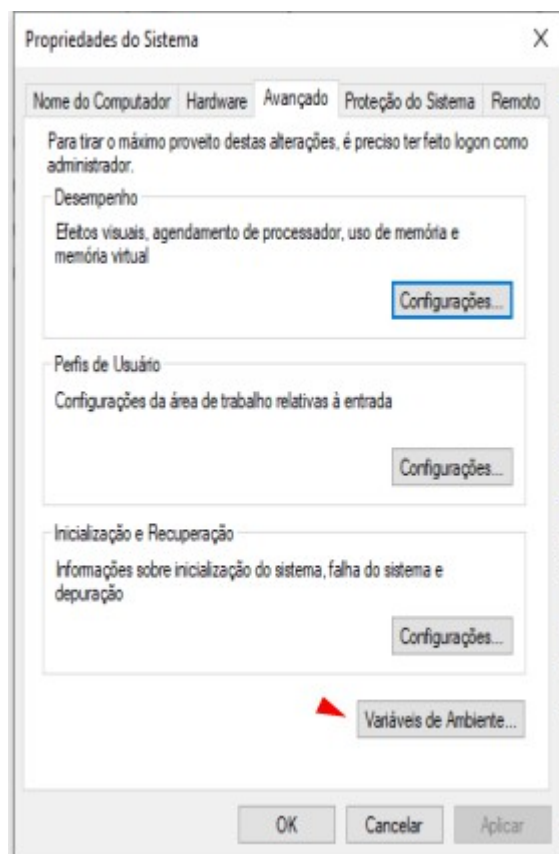
No Windows pesquise por “Sistema”



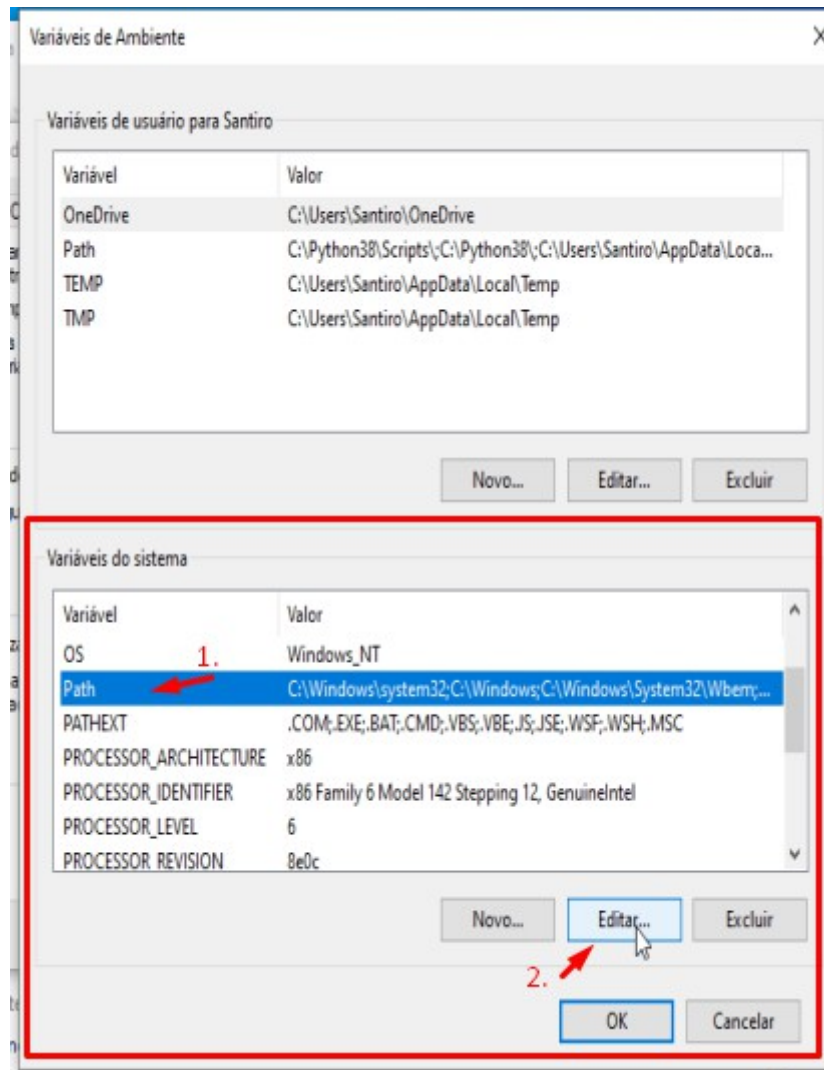
Agora vá em “Configurações avançadas do sistema”



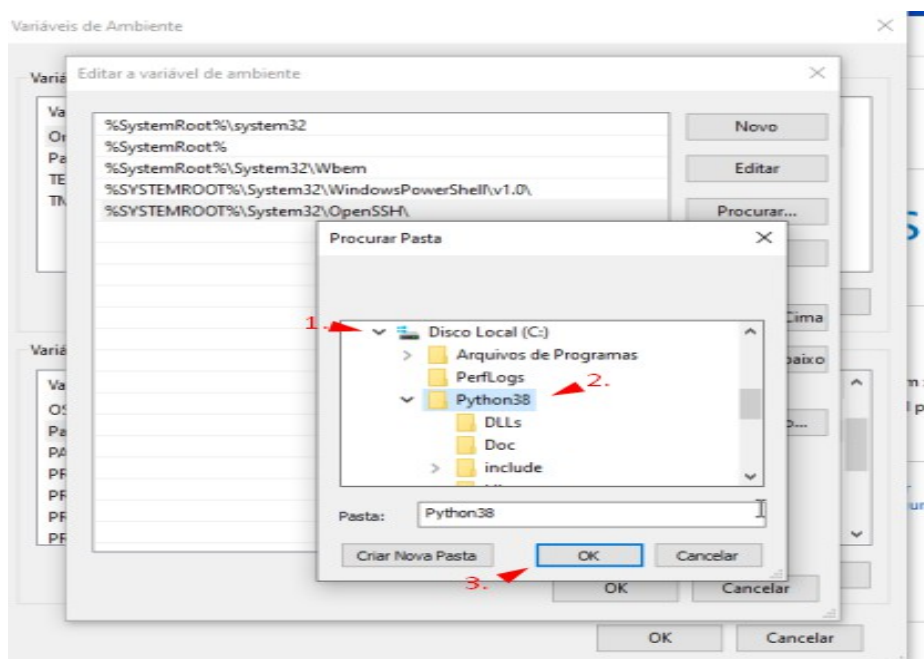
Selecione “Variáveis de Ambiente...”



Na coluna de “Variáveis do sistema” procure por “Path”, selecione a mesma e clique em “Editar...”



Na nova janela clique em “Procurar...”, será aberta uma janela onde precisaremos localizar a pasta onde o Python foi instalado, no meu caso eu instalei em “C:/Python38”



Feito isso, clique em “OK” em todas as janelas abertas. Pronto o Python já foi adicionado ao PATH.

```
C:\Users\Santiro>python
Python 3.8.1 (tags/v3.8.1:1b293b6, Dec 18 2019, 22:39:24) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> _
```

Instalando Scripts no Python através do “pip”

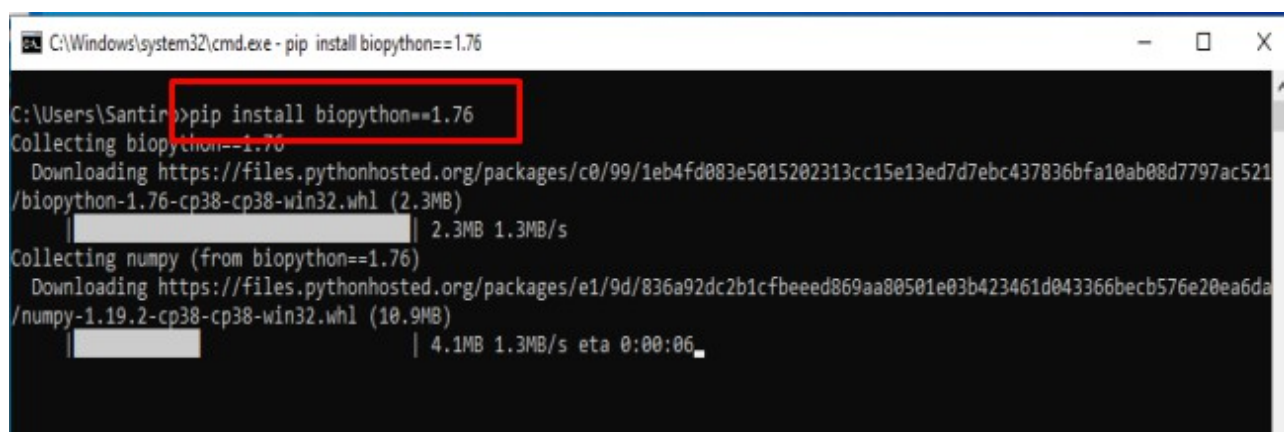
Em relação as dependências do nosso projeto, podemos destacar apenas duas: **BioPython** e **Django**. Ambas são dependências que precisamos instalar manualmente.

Instalando BioPython

Primeiramente inicie o terminal Windows, após iniciado digite seguinte comando:

```
pip install biopython==1.76
```

trabalharemos com a versão 1.76 pois foi a utilizada neste projeto.



```
C:\Windows\system32\cmd.exe - pip install biopython==1.76

C:\Users\Santiro>pip install biopython==1.76
Collecting biopython==1.76
  Downloading https://files.pythonhosted.org/packages/c0/99/1eb4fd083e5015202313cc15e13ed7d7ebc437836bfa10ab08d7797ac521/biopython-1.76-cp38-cp38-win32.whl (2.3MB)
    | 2.3MB 1.3MB/s
Collecting numpy (from biopython==1.76)
  Downloading https://files.pythonhosted.org/packages/e1/9d/836a92dc2b1cfbeed869aa80501e03b423461d043366becb576e20ea6da/numpy-1.19.2-cp38-cp38-win32.whl (10.9MB)
    | 4.1MB 1.3MB/s eta 0:00:06
```

O processo de instalação é automático, aguarde até receber a seguinte mensagem:

```
Successfully installed biopython-1.76 numpy-1.19.2
```

Caso apareça algum alerta para você em amarelo não se desespere, ele apenas estará alertando que existe uma versão mais atualizada do instalador de pacotes do Python, a escolha de upgrade é sua, para esse caso não irei me preocupar com a atualização.

```
WARNING: You are using pip version 19.2.3, however version 20.2.4 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.
```

Vamos testar se a instalação do BioPython foi realizada com sucesso com um simples exemplo disponível no próprio site do [BioPython](#).

Comece criando um arquivo qualquer com a extensão “.py”, no caso vou criar um “testeBIO.py” (Se atente onde irá criar o mesmo, por padrão vou deixar no Desktop), caso não consiga criar a extensão do arquivo como “.py” você precisará [desbloquear essa configuração no Windows](#).

Utilize o editor de texto de sua preferência, nós particularmente desenvolvemos todo o ambiente com o “[Visual Studio Code](#)” da Microsoft, cabe a você escolher o editor que mais lhe agrada, nesta VM utilizarei ferramentas nativas do Windows (vulgo bloco de notas, rs).

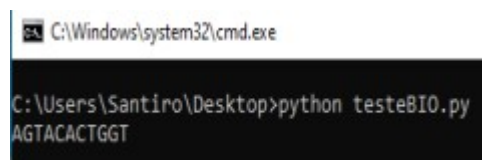
Insira as seguintes linhas no seu arquivo:

```
from Bio.Seq import Seq
my_seq = Seq("AGTACACTGGT")
print(my_seq)
```

Este arquivo irá fazer a impressão do DNA cadastrado porém alterando o seu tipo para “SEQ” o que referência que o mesmo é uma sequência. Para executar seu arquivo vá no mesmo diretório que o mesmo se encontra e digite:

```
python <nomedoarquivo.py>
```

No meu caso:



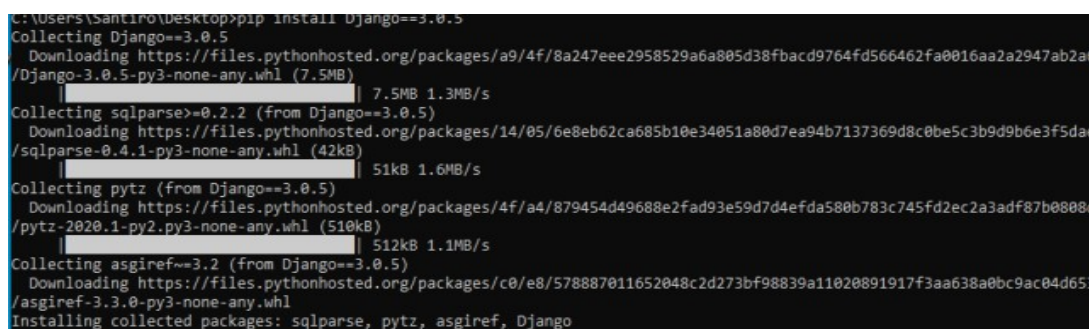
```
C:\Windows\system32\cmd.exe
C:\Users\Santiro\Desktop>python testeBIO.py
AGTACACTGGT
```

Obtendo essa saída tudo está ok! O BioPython foi instalado com sucesso. Vamos para a instalação do Django.

Instalação Django

O procedimento de instalação do Django é o mesmo que o do BioPython, basta usarmos o “pip” novamente, especificando a versão, no nosso caso, utilizaremos a versão 3.0.5 do Django, o comando fica da seguinte forma:

```
pip install Django==3.0.5
```



```
C:\Users\Santiro\Desktop>pip install Django==3.0.5
Collecting Django==3.0.5
  Downloading https://files.pythonhosted.org/packages/a9/4f/8a247eee2958529a6a805d38fbacd9764fd566462fa0016aa2a2947ab2a/Django-3.0.5-py3-none-any.whl (7.5MB)
    | 7.5MB 1.3MB/s
Collecting sqlparse>=0.2.2 (from Django==3.0.5)
  Downloading https://files.pythonhosted.org/packages/14/05/6e8eb62ca685b10e34051a80d7ea94b7137369d8c0be5c3b9d9b6e3f5da/sqlparse-0.4.1-py3-none-any.whl (42kB)
    | 51kB 1.6MB/s
Collecting pytz (from Django==3.0.5)
  Downloading https://files.pythonhosted.org/packages/4f/a4/879454d49688e2fad93e59d7d4efda580b783c745fd2ec2a3adf87b0808/pytz-2020.1-py2.py3-none-any.whl (510kB)
    | 512kB 1.1MB/s
Collecting asgiref~3.2 (from Django==3.0.5)
  Downloading https://files.pythonhosted.org/packages/c0/e8/578887011652048c2d273bf98839a11020891917f3aa638a0bc9ac04d65/asgiref-3.3.0-py3-none-any.whl
Installing collected packages: sqlparse, pytz, asgiref, Django
```

Aguarde até a conclusão da instalação...

```
Successfully installed Django-3.0.5 asgiref-3.3.0 pytz-2020.1 sqlparse-0.4.1
```

Para verificarmos a versão que foi instalada digite o seguinte comando:

```
python -m django --version
```

Se a resposta for 3.0.5, tudo está correto!

```
C:\Users\Santiro\Desktop>python -m django --version  
3.0.5
```

Instalando softwares externos (Bioinformática)

Após realizar as instalações e configurações de ambiente em Python, vamos para a próxima etapa que é a configuração de softwares de ambientes, ou melhor dizendo, softwares da Bioinformática. Estes softwares são utilizados em 2 módulos do nosso projeto, os módulos de modelagem 3D e no módulo de comparar duas sequências.

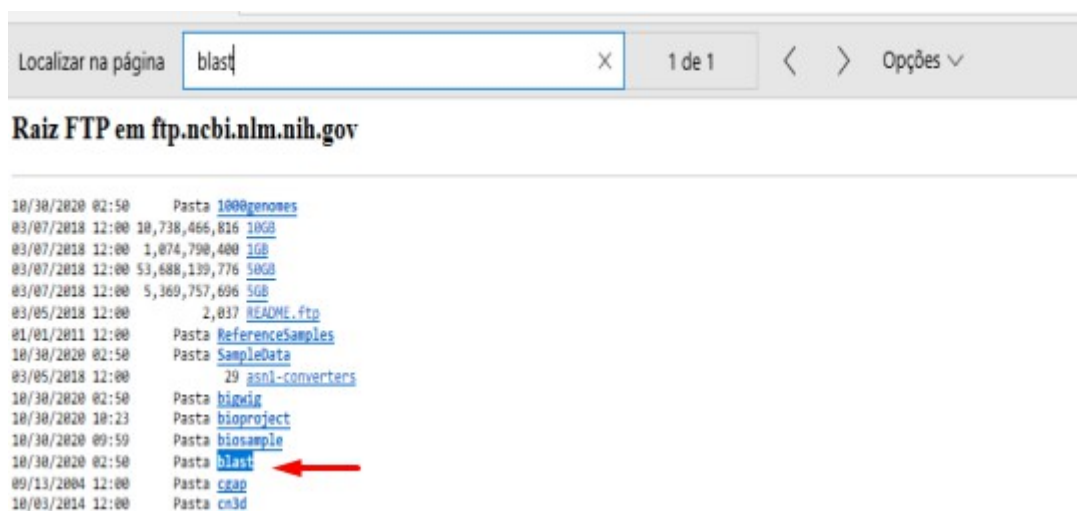
Primeiro vamos instalar as depêndencias do módulo de comparar sequências, nele precisaremos instalar e configurar o BLASTP, uma ferramenta para gerar o alinhamento entre duas sequências e nos dar algumas informações desejadas.

Instalando o Blast

O primeiro passo é entrar no FTP do NCBI, onde temos a opção de diversas ferramentas e bases de dados para Download.

[Link raiz do FTP – NCBI](#)

Ao entrar no FTP, pesquise pelo diretório “blast”.



Agora clique em “Executables”.



Para o diretório de nível superior

12/16/2019 12:00	Pasta	WGS_TOOLS
05/10/2004 12:00		20,087 blastftp.txt
10/13/2020 09:57	Pasta	db
11/05/2019 12:00	Pasta	demo
08/24/2020 09:04	Pasta	documents
10/24/2019 12:00	Pasta	executables
11/13/2019 12:00	Pasta	matrices
10/27/2020 05:54	Pasta	temp
10/01/2020 11:25	Pasta	windowmasker_files

blast+.

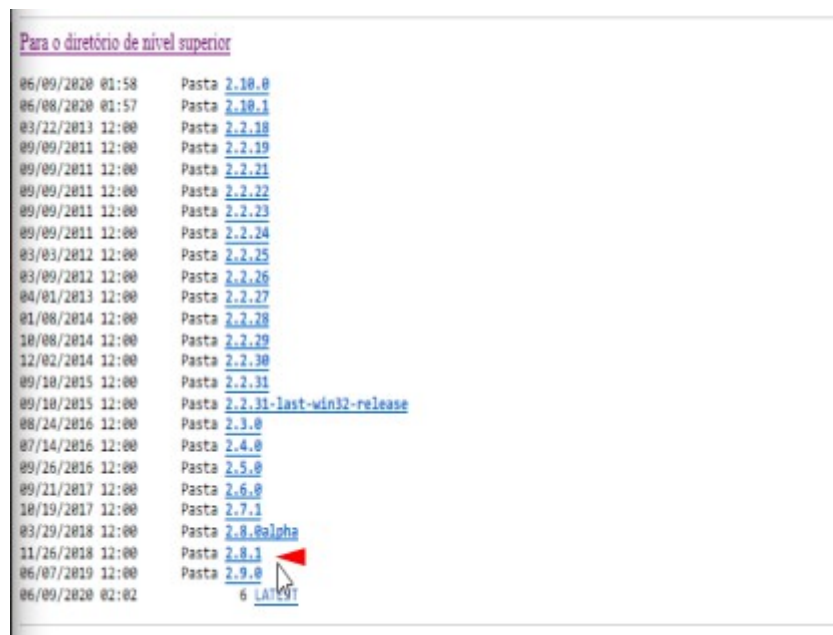


Diretório FTP /blast/executables/ em ftp.ncbi.nlm.nih.gov

Para o diretório de nível superior

07/12/2011 12:00		13 LATEST
06/09/2020 02:02	Pasta	blast+
05/25/2018 12:00	Pasta	iblast
11/13/2018 12:00	Pasta	legacy.NOTSUPPORTED
08/21/2019 12:00	Pasta	magicblast
11/07/2019 12:00	Pasta	remote-fuser
02/27/2013 12:00	Pasta	rmbblast

entre no diretório 2.8.1.



Para o diretório de nível superior

06/09/2020 01:58	Pasta	2.10.0
06/08/2020 01:57	Pasta	2.10.1
03/12/2013 12:00	Pasta	2.2.18
09/09/2011 12:00	Pasta	2.2.19
09/09/2011 12:00	Pasta	2.2.21
09/09/2011 12:00	Pasta	2.2.22
09/09/2011 12:00	Pasta	2.2.23
09/09/2011 12:00	Pasta	2.2.24
03/03/2012 12:00	Pasta	2.2.25
03/09/2012 12:00	Pasta	2.2.26
04/01/2013 12:00	Pasta	2.2.27
01/08/2014 12:00	Pasta	2.2.28
10/08/2014 12:00	Pasta	2.2.29
12/02/2014 12:00	Pasta	2.2.30
09/10/2015 12:00	Pasta	2.2.31
09/10/2015 12:00	Pasta	2.2.31-last-win32-release
08/24/2016 12:00	Pasta	2.3.0
07/14/2016 12:00	Pasta	2.4.0
09/26/2016 12:00	Pasta	2.5.0
09/21/2017 12:00	Pasta	2.6.0
10/19/2017 12:00	Pasta	2.7.1
03/29/2018 12:00	Pasta	2.8.0alpha
11/26/2018 12:00	Pasta	2.8.1
06/07/2019 12:00	Pasta	2.9.0
06/09/2020 02:02		6 LATEST

e clique em “ncbi-blast-2.8.1+-win64.exe” (se seu windows for 64 bits) para efetuar o Download do instalador. Como o meu windows na máquina virtual é de 32 bits, vou utilizar outro endereço para o download, os 2 links estarão disponíveis.

Download citado logo abaixo:

Para quem se perdeu, segue o caminho relativo do FTP (Windows 64 bits) => [FTP](#).

Para o diretório de nível superior

11/26/2018 12:00	85	ChangeLog
11/26/2018 12:00	28,183,281	ncbi-blast-2.8.1+-2-src.rpm
11/26/2018 12:00	62	ncbi-blast-2.8.1+-2-src.rpm.md5
11/26/2018 12:00	194,792,784	ncbi-blast-2.8.1+-2.x86_64.rpm
11/26/2018 12:00	65	ncbi-blast-2.8.1+-2.x86_64.rpm.md5
11/26/2018 12:00	24,973,554	ncbi-blast-2.8.1+-src.tar.gz
11/26/2018 12:00	63	ncbi-blast-2.8.1+-src.tar.gz.md5
11/26/2018 12:00	28,874,837	ncbi-blast-2.8.1+-src.zip
11/26/2018 12:00	68	ncbi-blast-2.8.1+-src.zip.md5
11/26/2018 12:00	94,442,276	ncbi-blast-2.8.1+-win64.exe
11/26/2018 12:00	62	ncbi-blast-2.8.1+-win64.exe.md5
11/26/2018 12:00	241,992,963	ncbi-blast-2.8.1+-x64-linux.tar.gz
11/26/2018 12:00	69	ncbi-blast-2.8.1+-x64-linux.tar.gz.md5
11/26/2018 12:00	134,348,474	ncbi-blast-2.8.1+-x64-macosx.tar.gz
11/26/2018 12:00	78	ncbi-blast-2.8.1+-x64-macosx.tar.gz.md5
11/26/2018 12:00	94,159,828	ncbi-blast-2.8.1+-x64-win64.tar.gz
11/26/2018 12:00	69	ncbi-blast-2.8.1+-x64-win64.tar.gz.md5
11/26/2018 12:00	136,049,938	ncbi-blast-2.8.1+-dmg
11/26/2018 12:00	56	ncbi-blast-2.8.1+-dmg.md5

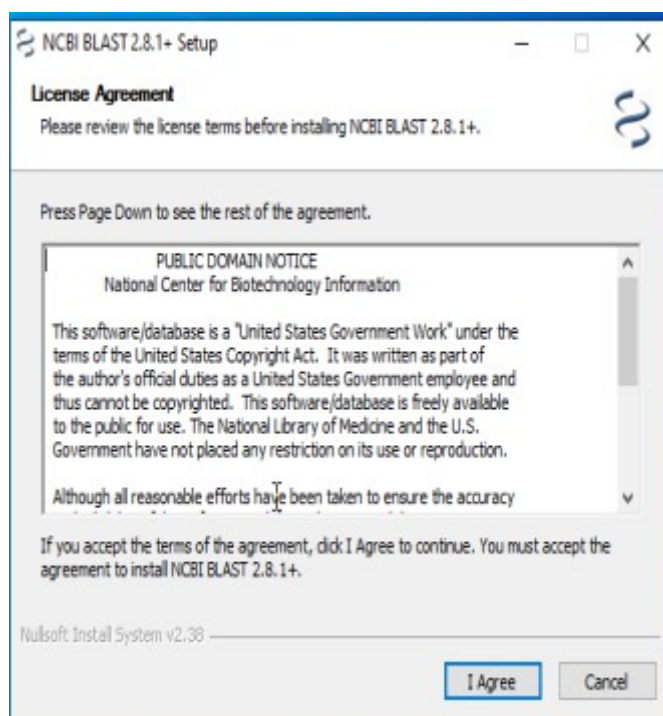
Para quem estiver usando o Windows 32bits, segue o LINK do Download => [FTP](#)

Index of /blast/executables/blast+/2.2.31-last-win32-release

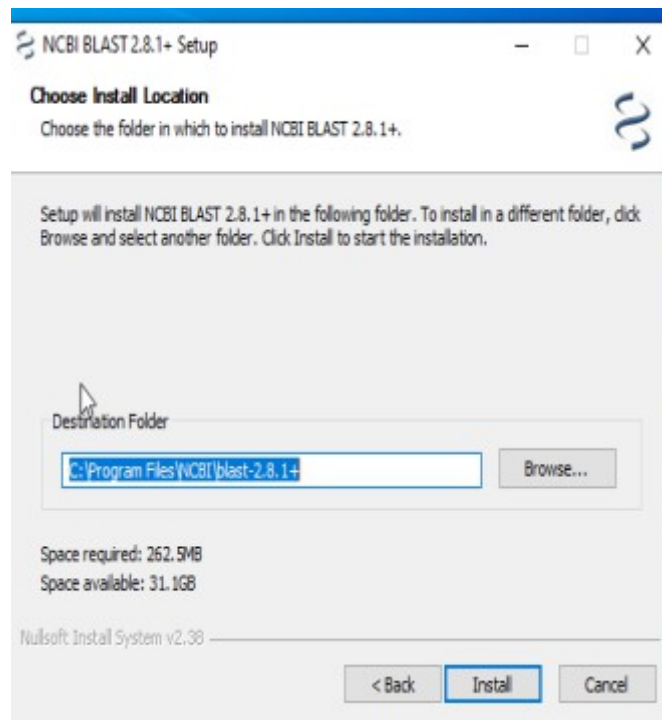
Name	Last modified	Size
Parent Directory		-
ncbi-blast-2.2.31+ia32-win32.tar.gz	2015-06-15 08:48	60M
ncbi-blast-2.2.31+ia32-win32.tar.gz.md5	2015-06-15 08:48	71
ncbi-blast-2.2.31+win32.exe	2015-06-15 08:49	60M
ncbi-blast-2.2.31+win32.exe.md5	2015-06-15 08:49	63

Após o Download do “.exe” o processo de instalação é comum:

Selecione “I agree” na primeira janela.



No local de instalação, também deixo por padrão, clique em “Install”.



Aguarde o final da instalação e clique em “Close”.

Com isso já temos o Blast instalado, ele por padrão já adiciona uma variável de ambiente para utilização via terminal, então já podemos passar ao próximo software.

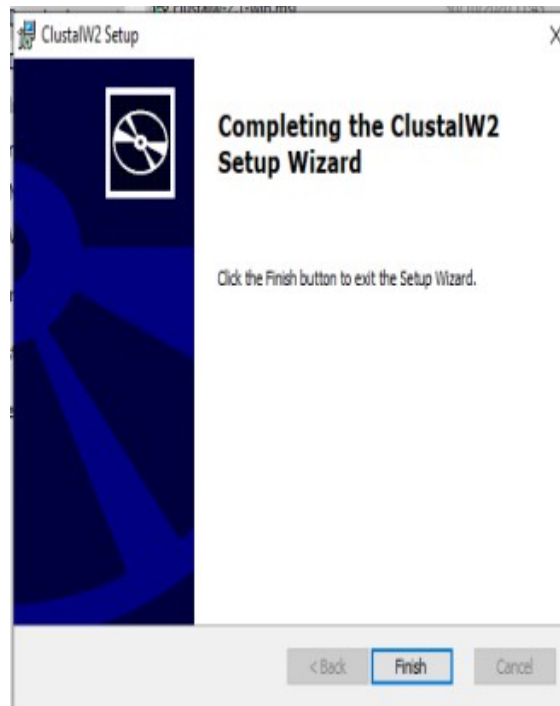
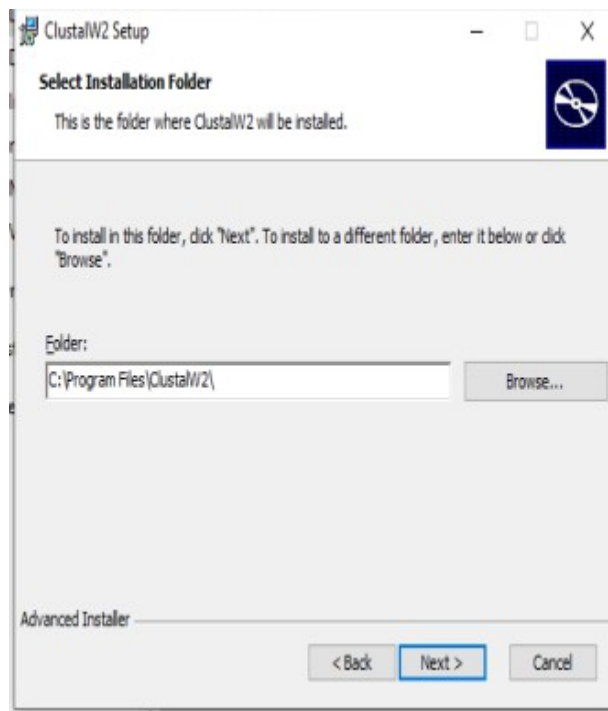
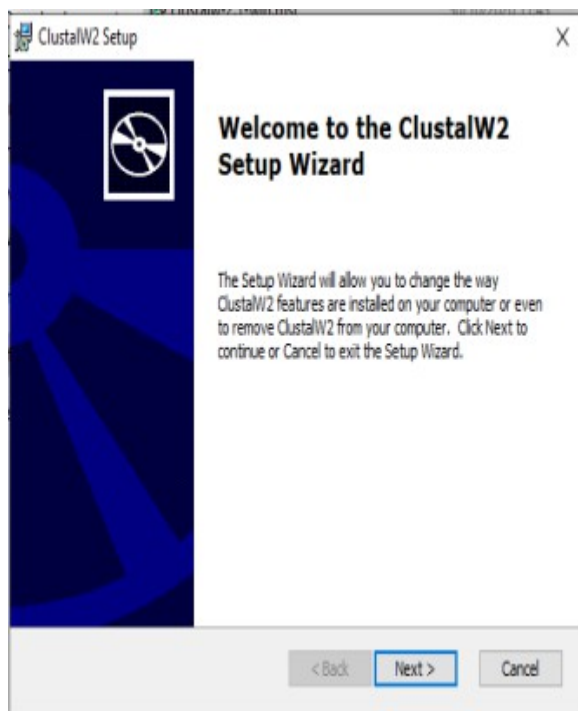
Instalando o Clustaw

O Clustaw também é uma ferramenta de alinhamento, no entanto a mesa oferece uma comparação muito mais detalhada, o que é ideal para a montagem de um modelo de proteína 3D a partir de uma referência (template).

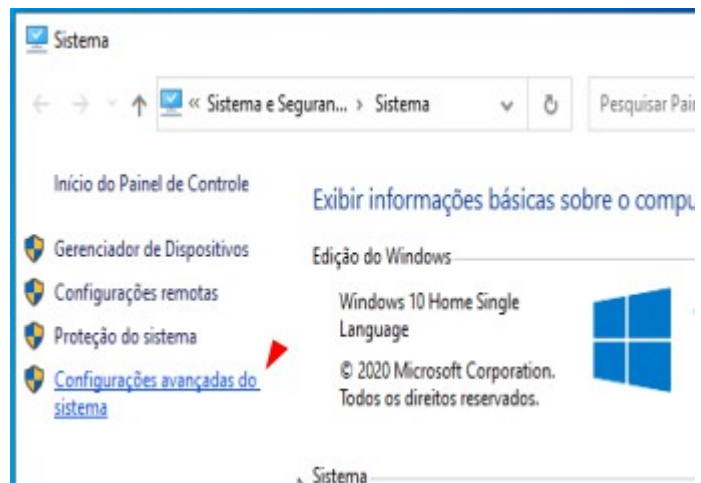
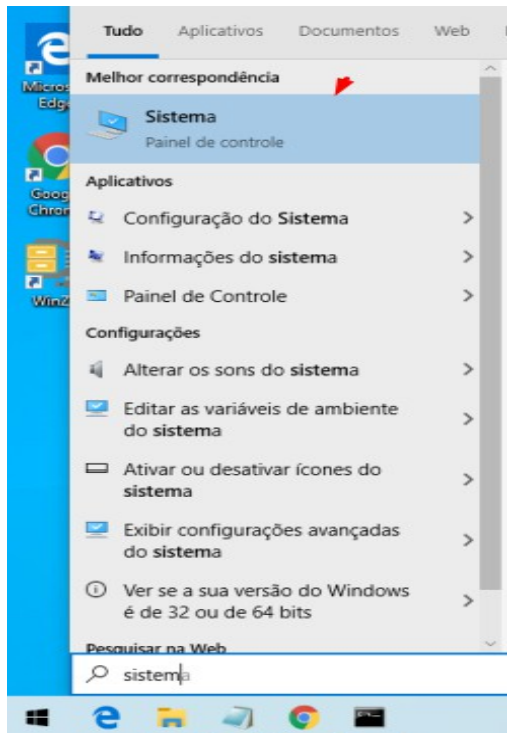
Acesse este link para ir diretamente ao repositório do FTP do ClustalW => [FTP](#).



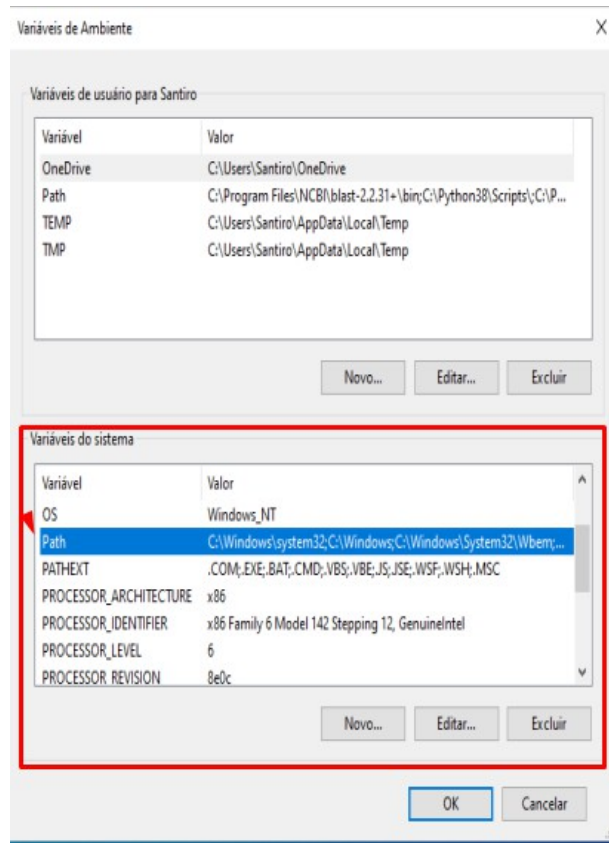
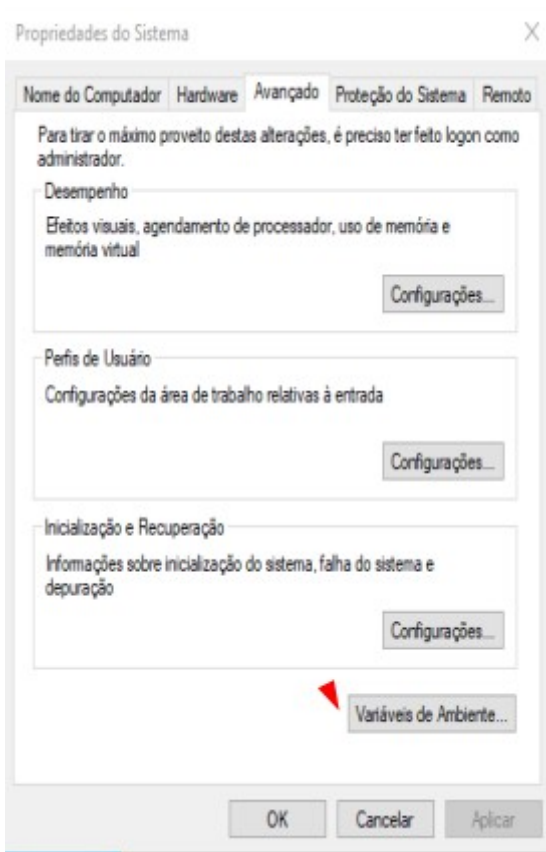
Após realizar o Download do arquivo “.msi” vamos instalar o programa. Neste caso o instalador de pacotes é muito simples, aparecerão 2 janelas, sendo necessário clicar em “Next” => “Next” => “Install” => “Finish” na última.



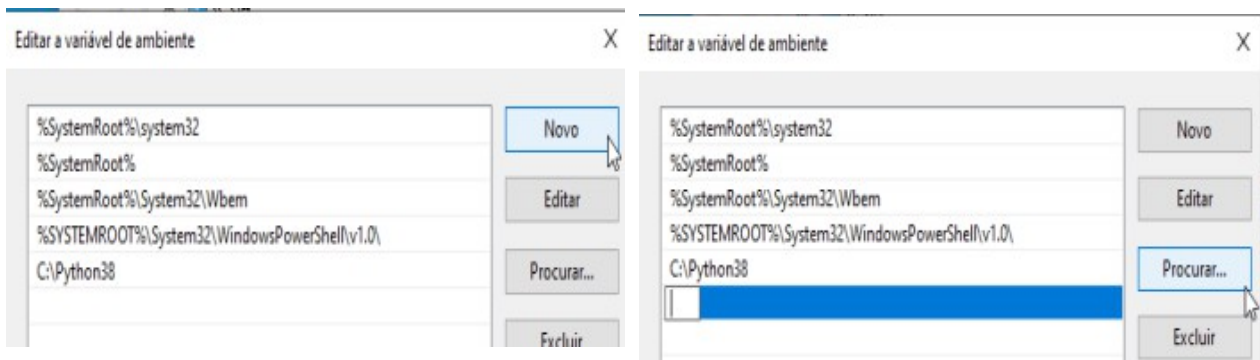
Feito a instalação vamos instalar o Clustalw no PATH do Windows. Para isto, novamente pesquisaremos por “Sistema” e entraremos nas “Configurações avançadas do sistema”.



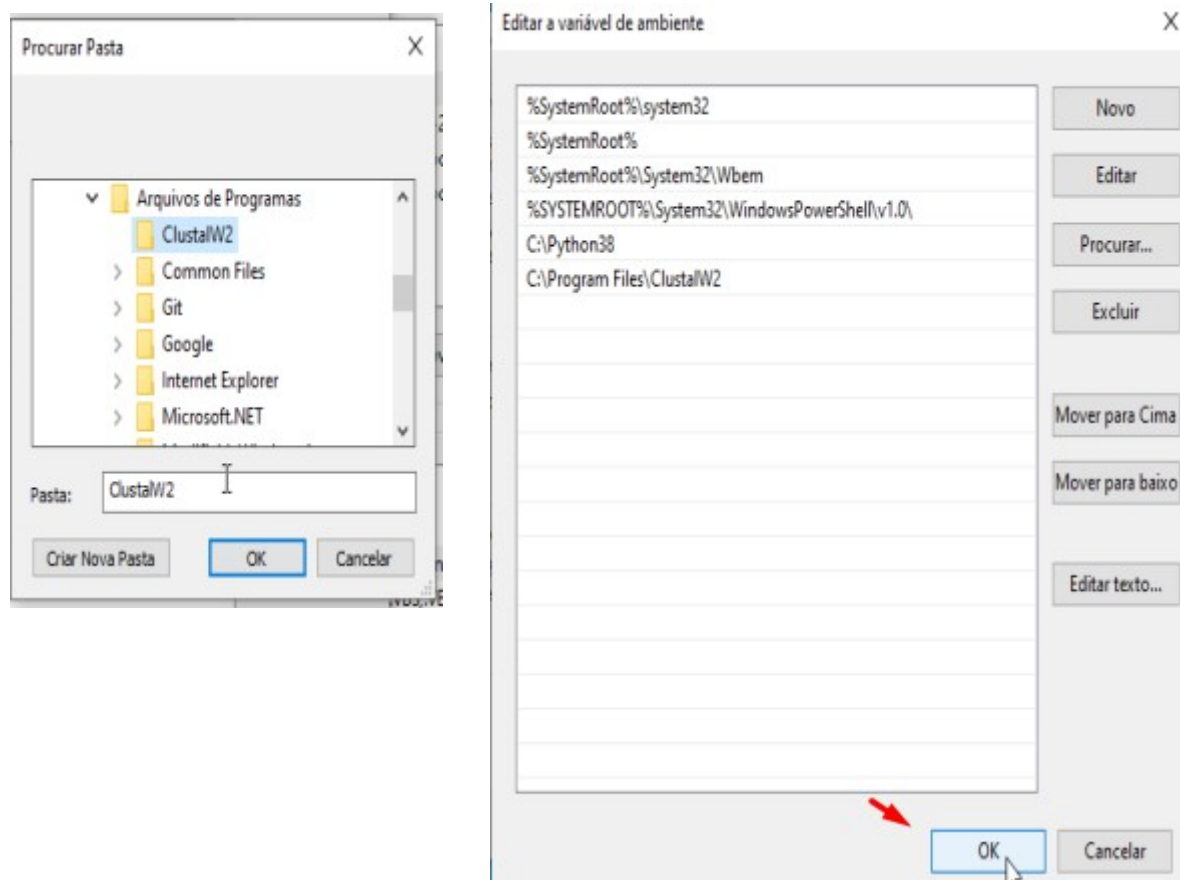
Clique em “Variáveis de Ambiente...” e localize dentro de “Variáveis do sistema” a linha “PATH”



Clique em “Editar” para abrir uma nova janela, depois em “Novo” => “Procurar...” selecionando onde o Clustaw foi instalado.



Selecionando o diretório em que o Clustaw foi instalado:



Feito estas etapas clique em “OK” para todas as janelas e a variável de ambiente já estará configurada e acessível pelo nosso projeto. Proximo software “MODELLER”

Instalando e configurando o Modeller

O Modeller é o último software utilizado para a rotina de modelar uma proteína 3d por homologia, ele é o responsável por gerar os modelos 3d a partir da análise que todos os softwares anteriores criarem. O Modeller é desenvolvido em Fortran 90, utilizando Python como linguagem principal para o controle.

Para o processo de instalação do Modeller nós precisamos de uma licença, a mesma só é disponibilizada através do cadastro de um E-mail em seu site oficial, por questões de respeito a licença e ao projeto, não disponibilizarei a “Chave de ativação” do produto, mostrarei como adquirir através do cadastro.

Primeiramente vamos no site oficial do modeller, dentro do ambiente da Salilab, acesse este [LINK](#) Novamente escolha a sua versão do sistema operacional e baixe o instalador, porém na mesma página note que é avisado sobre a necessidade do registro de licença para a utilizar o software e no mesmo parágrafo é disponibilizado o [LINK](#) para o registro.

9. This license shall be for a term of 5 years except that upon any breach of this Agreement by LICENSEE, LICENSOR shall have the right to terminate this license immediately upon notice to LICENSEE.

Name:
Title:
(e.g. Dr., Mr., Mrs.)
Institution:
(no abbreviations, please)
Street Address:
Email:
Confirm email:
☒ Notify me by email of new MODELLER releases

On which platforms do you want to use MODELLER? (Check all that apply.)

☐ I want to use with a web service (e.g. [ModVeb](#), [ModLoop](#)) or a program (e.g. [Chimera](#))
☐ Microsoft Windows ☐ Apple Mac OS X ☐ Linux (32 bit PC) ☐ Linux (x86_64 machine, e.g. Opteron)
☐ Linux (32-bit ARM, e.g. Raspberry Pi) ☐ Linux (64-bit ARM, e.g. Raspberry Pi) ☐ IBM AIX ☐ FreeBSD
☐ Other(s) (please specify)

Guidelines for email addresses:

1. Provide a valid email address. The Modeller license key will be sent to the address you provide. If you give an incorrect address, you will not receive the key.
2. Provide only one address. The license key will not be sent to multiple addresses.
3. Give your official academic email address rather than a home email address. The server tries to use your email address to determine your academic status. (It is not just a simple check for a '.edu' extension, but it cannot recognize every institution.) If you do not provide an academic email address, your request must be processed manually, which will result in significant delays, and you may be denied a license.

Please note that your email address is used by us only for Modeller. It will only be used to send your license key, and (unless you opted out above) to notify you of any new Modeller releases.

Realize o cadastro preenchendo os campos necessários. Feito isso será disponibilizada a chave de ativação do produto no seu E-mail.

☐ ★ Modeller License Se. Modeller license key - for the MODELLER program. The MODELLER license key is M...

Modeller License Server <license@salilab.org>
para mim, modeller-care

🌐 inglês > português Traduzir mensagem

Hi,

Thank you very much for signing the license agreement for the **MODELLER** program.

The **MODELLER** license key is

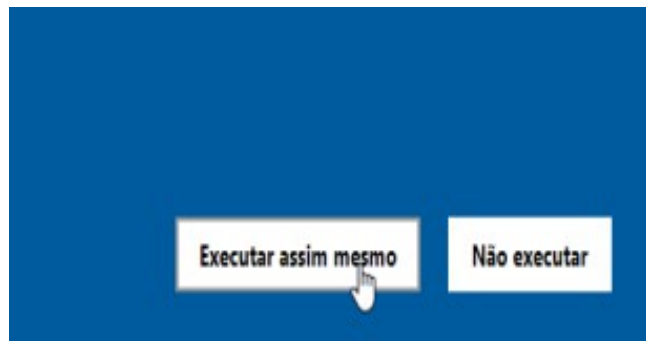
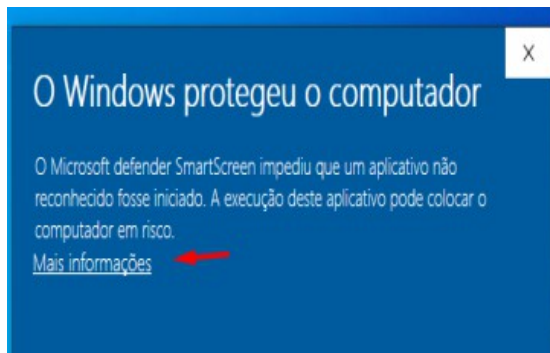
This license key will work for any release of **MODELLER** 8 or 9 (e.g. 9.24, 8v2) and should be given to the **MODELLER** installer when requested.

Please keep this email for reference, in case you want to install **MODELLER** on a different platform or computer in future (the key is the same for all platforms). If you lose the email, however, you can always fill in the license agreement again.

Regards,

Ben Webb, **Modeller** Caretaker

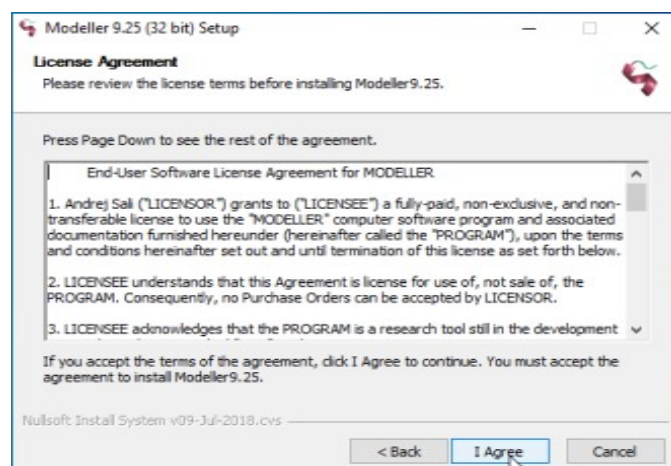
Agora com a chave de ativação podemos dar continuidade a instalação do software. Caso apareça uma janela alertando que o Windows impediu um aplicativo desconhecido, clique em “Mais informações” e será liberada a opção “Executar assim mesmo”



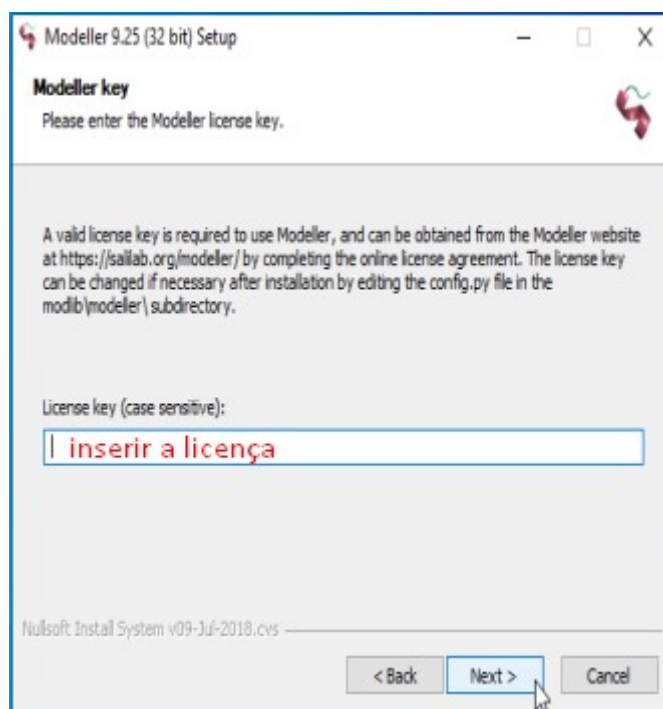
A primeira janela vamos clicar em “Next >”



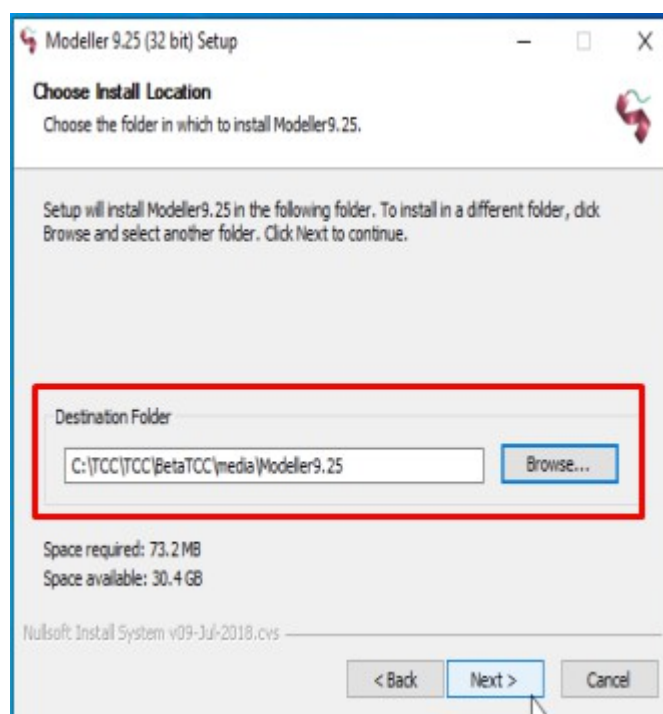
Em seguida “I Agree”



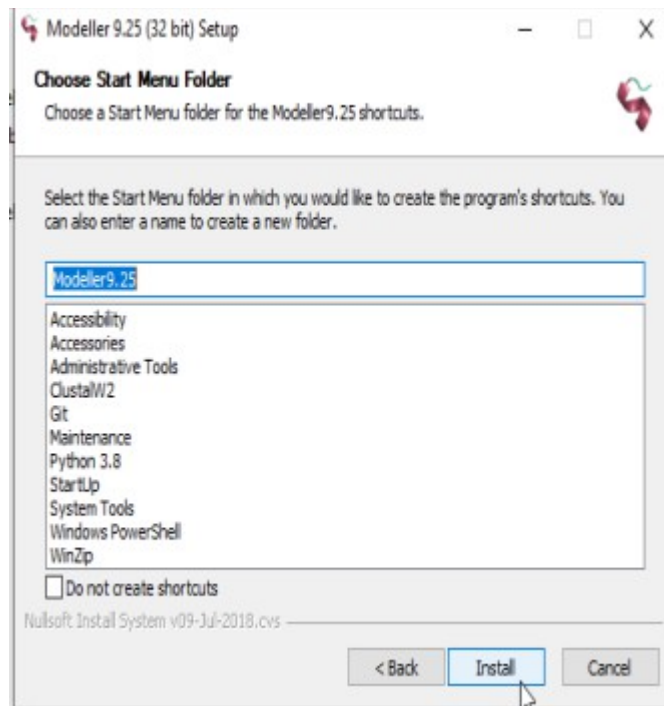
A terceira janela pede a “Chave de Licença” enviada no seu E-mail, note que este campo é “Case sensitive” então precisamos colocar exatamente como no E-mail.



Na quarta janela precisamos mudar as configurações para o Modeller ser instalado dentro do nosso Projeto! Para isso no campo “Destination Folder” precisaremos inserir o caminho do diretório “media”.



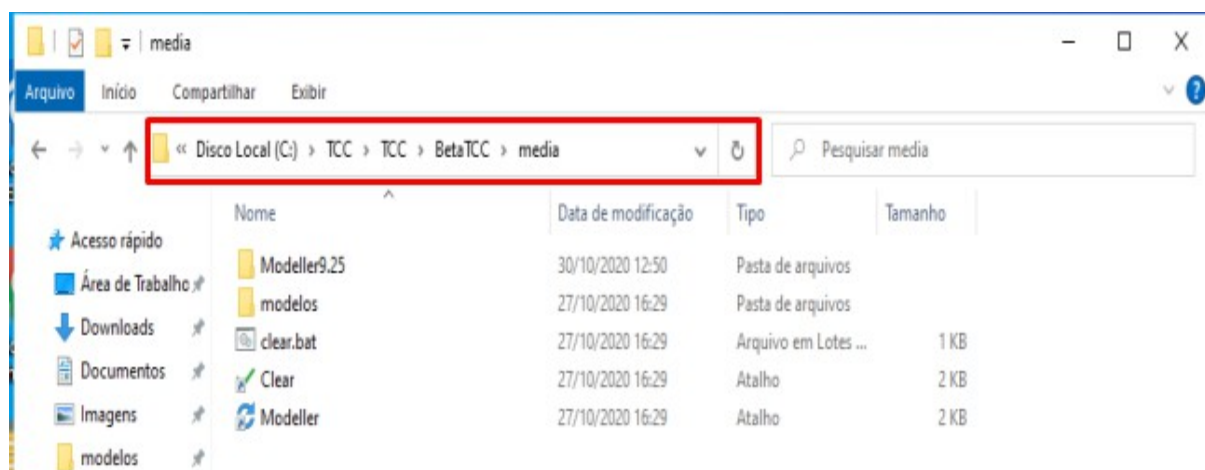
Na quinta janela clique em “Install”



Após a conclusão do instalador, finalize clicando em “Finish”. Até aqui tudo ok, já instalamos o Modeller, porém ainda não terminamos a configuração para que o mesmo funcione nas chamadas do nosso Projeto. Vamos então começar a configurar o Modeller.

Configurando o Modeller para o Projeto

Primeiro vamos para o local onde o modeller foi instalado, o diretório “media” dentro do Projeto.



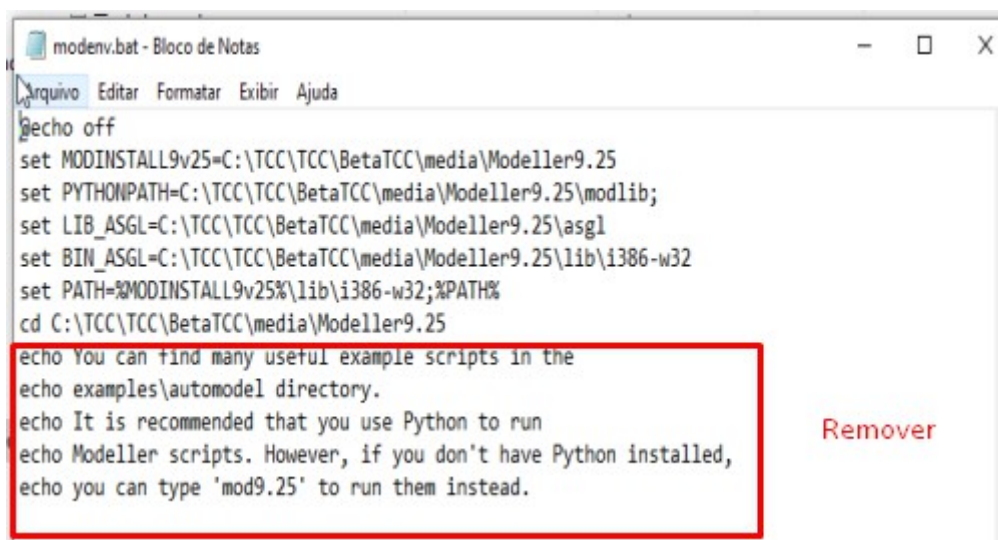
Note que por padrão temos 3 arquivos já criados neste diretório, sendo 1 arquivo “.bat” e 2 arquivos “atalhos”.

clear.bat	27/10/2020 16:29	Arquivo em Lotes ...	1 KB
Clear	27/10/2020 16:29	Atalho	2 KB
Modeller	27/10/2020 16:29	Atalho	2 KB

Eles estão aqui propositalmente, no entanto vou mostrar a configuração do zero, já que toda a configuração depende de máquina para máquina. Mas antes vamos terminar de configurar o Modeller.

Entre no diretório do Modeller => bin e procure por um arquivo “modenv.bat”, este é o arquivo responsável pelas definições e chamadas das funções necessárias para o desenvolvimento do modelo 3d. O arquivo pode variar de acordo com o seu sistema operacional, versão do modeller instalada, tipo, local de instalação, etc.. O importante neste arquivo é o apontamento estar correto (instalado dentro do diretório media). Vamos modificar somente o final deste “.bat” para funcionar com a chamada do nosso Projeto.

Na primeira vez que for aberto, provavelmente ele estará parecido com esta versão, é importante removermos as linhas destacadas na imagem logo abaixo.



```
modenv.bat - Bloco de Notas
Arquivo  Editar  Formatar  Exibir  Ajuda
@echo off
set MODINSTALL9v25=C:\TCC\TCC\BetaTCC\media\Modeller9.25
set PYTHONPATH=C:\TCC\TCC\BetaTCC\media\Modeller9.25\modlib;
set LIB_ASGL=C:\TCC\TCC\BetaTCC\media\Modeller9.25\asgl
set BIN_ASGL=C:\TCC\TCC\BetaTCC\media\Modeller9.25\lib\i386-w32
set PATH=%MODINSTALL9v25%\lib\i386-w32;%PATH%
cd C:\TCC\TCC\BetaTCC\media\Modeller9.25
echo You can find many useful example scripts in the
echo examples\automodel directory.
echo It is recommended that you use Python to run
echo Modeller scripts. However, if you don't have Python installed,
echo you can type 'mod9.25' to run them instead.
```

Substitua pelas seguintes linhas:

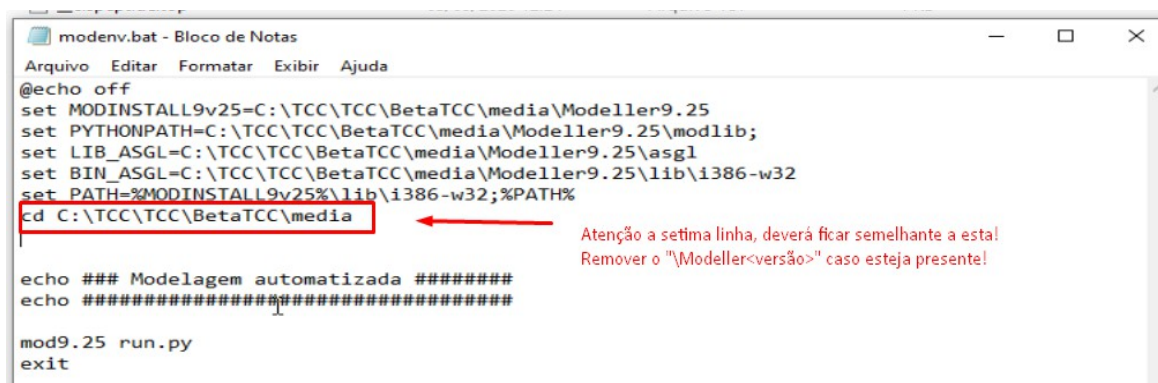
```
echo ##### Modelagem Automatizada #####
echo #####
```

```
mod9.24 run.py
exit
```

Também é necessário remover o final da 7 Linha “Modeller9.25”, deixando somente:

cd C:\TCC\TCC\BETATCC\media

Ficará da seguinte forma:



```
modenv.bat - Bloco de Notas
Arquivo  Editar  Formatar  Exibir  Ajuda
@echo off
set MODINSTALL9v25=C:\TCC\TCC\BetaTCC\media\Modeller9.25
set PYTHONPATH=C:\TCC\TCC\BetaTCC\media\Modeller9.25\modlib;
set LIB_ASGL=C:\TCC\TCC\BetaTCC\media\Modeller9.25\asgl
set BIN_ASGL=C:\TCC\TCC\BetaTCC\media\Modeller9.25\lib\i386-w32
set PATH=%MODINSTALL9v25%\lib\i386-w32;%PATH%
cd C:\TCC\TCC\BetaTCC\media

echo ### Modelagem automatizada #####
echo #####

mod9.25 run.py
exit
```

obs. Se atente a sua versão do Modeller

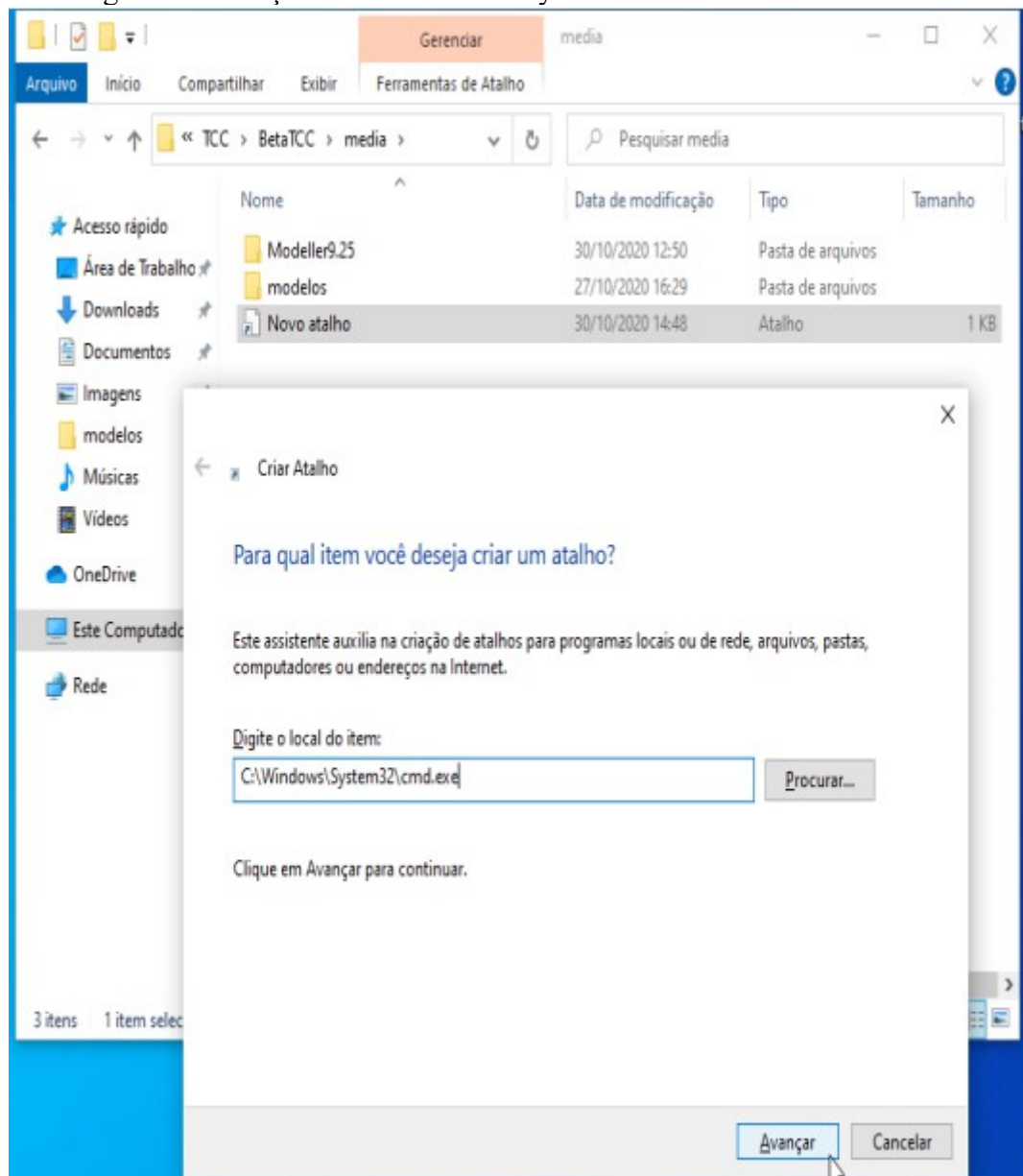
Pronto! O Modeller já está funcionando, Agora vamos configurar os 3 arquivos no “media”.
Começaremos excluindo os 3 arquivos

clear.bat	27/10/2020 16:29	Arquivo em Lotes ...	1 KB
Clear	27/10/2020 16:29	Atalho	2 KB
Modeller	27/10/2020 16:29	Atalho	2 KB

Após a exclusão, vamos começar a montar o Modeller.lnk:

Criando e configurando o Modeller.lnk

Primeiramente, clique com o botão direito do mouse e selecione a opção “novo” => “atalho”.
No campo “Digite o local do item” coloque o endereço do “cmd.exe”, geralmente ele fica localizado no seguinte endereço: “C:\WINDOWS\system32\cmd.exe”



Altere o Nome para “Modeller”

Digite um nome para o atalho:

Agora clique com o botão direito em cima do atalho criado e vá em “propriedades”.

No campo “Destino” já temos escrito: “C:\WINDOWS\system32\cmd.exe” agora vamos adicionar o endereço até o caminho do arquivo “.bat” do modeller que havíamos editado com um parametro “/K” antes, no meu caso o caminho é:

"C:\TCC\TCC\BetaTCC\media\Modeller9.25\bin\modenv.bat"

Agora adicionamos o “/K” antes ficando:

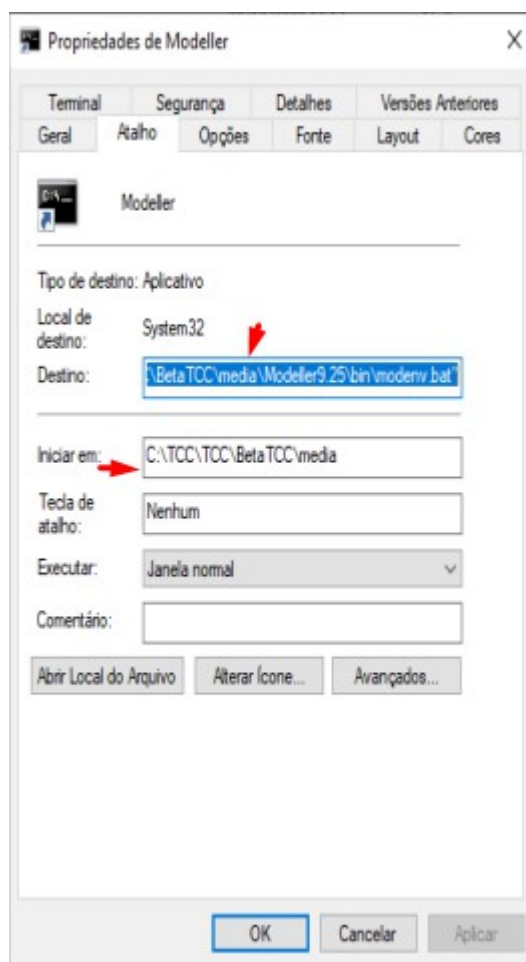
/K "C:\TCC\TCC\BetaTCC\media\Modeller9.25\bin\modenv.bat"

E por fim “concatenamos” ao comando anterior, ficará desta forma:

C:\WINDOWS\system32\cmd.exe /K "C:\TCC\TCC\BetaTCC\media\Modeller9.25\bin\modenv.bat"

Adicione o caminho dentro do campo de “Destino”

E no campo “Iniciar em” escreva C:\TCC\TCC\BetaTCC\media\



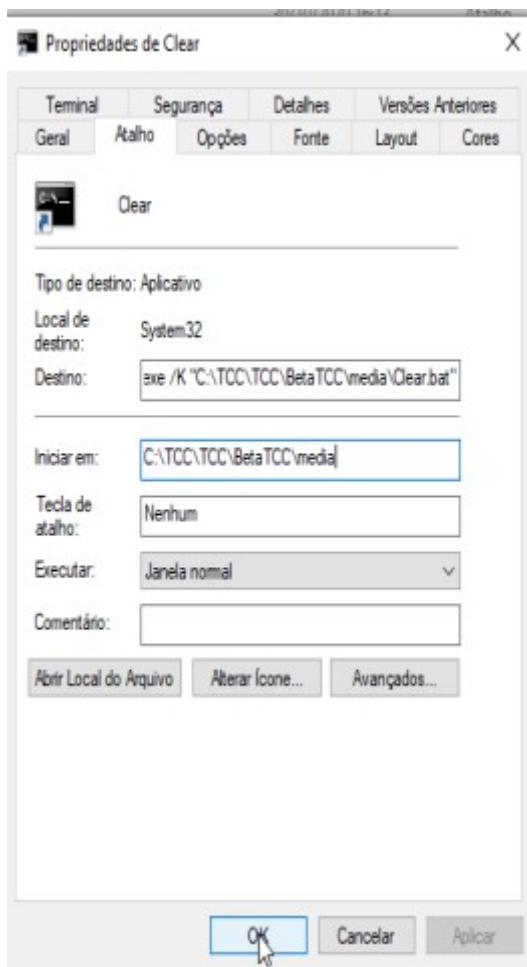
Clique em “Aplicar” e “OK”. Pronto, o “Modeller.lnk” já está configurado, agora vamos para o “Clear.lnk”

Criando e configurando o “Clear.lnk”

Novamente o mesmo processo, comece criando um novo atalho apontando para o “cmd.exe” igual o processo de criação do “Modeller.lnk”, depois em propriedades no campo de “Destino” adicionaremos o comando sublinhado a frente do que já estava escrito (**em negrito**):

C:\Windows\System32\cmd.exe /K "C:\TCC\TCC\BetaTCC\media\Clear.bat"

E no campo “Iniciar em” escreva C:\TCC\TCC\BetaTCC\media\



Feito isso temos tudo configurado, o arquivo “Clear.bat” é gerado automaticamente e sempre atualizado dinamicamente através dos Scripts do nosso projeto, então não precisamos nos preocupar com ele.

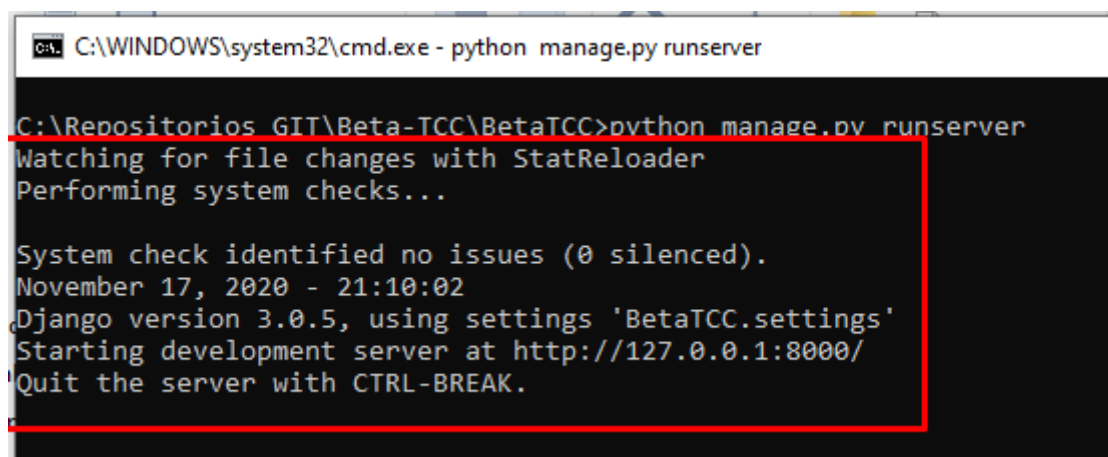
Como iniciar o servidor local

Para iniciar o servidor local abra o “Terminal”, vá até o diretório do projeto que contenha o arquivo “manage.py”.

Nome	Data de modificação	tipo	tamanho
BetaTCC	06/07/2020 15:27	Pasta de arquivos	
funcoes	30/07/2020 17:52	Pasta de arquivos	
media	17/11/2020 19:59	Pasta de arquivos	
Modelagem	19/10/2020 18:53	Pasta de arquivos	
pipeline	19/10/2020 20:15	Pasta de arquivos	
db.sqlite3	17/11/2020 19:59	Arquivo SQLITE3	172 KB
manage.py	06/07/2020 15:25	Arquivo PY	1 KB
Modeller	26/07/2020 21:30	Atalho	1 KB

E digite o seguinte comando no terminal:

```
python manage.py runserver
```



```
C:\WINDOWS\system32\cmd.exe - python manage.py runserver

C:\Repositorios GIT\Beta-TCC\BetaTCC>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
November 17, 2020 - 21:10:02
Django version 3.0.5, using settings 'BetaTCC.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Ao aparecer esta mensagem, o servidor estará rodando no endereço “<http://127.0.0.1:8000/>”. Para visualizar é necessário somente colar o endereço no seu navegador.

Apresentando o Projeto

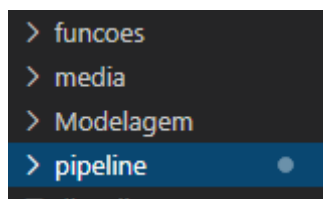
O Projeto é dividido em 3 módulos principais: Cadastro de DNA / Analises, Comparar duas sequências de Proteínas e Gerar um Modelo 3d de uma Proteína por homologia.

Nos arquivos cada um deles é representado por um “Diretório” sendo:

Modelagem é equivalente as funções de **Cadastro de DNA / Analises**.

Funcoes é equivalente as funções de **Comparar duas sequências de Proteínas**.

Pipeline é equivalente as funções para **Gerar um Modelo 3d de uma Proteína por homologia**.



Media é um diretório responsável por coletar e armazenar os arquivos gerados.

Modulo de Cadastro / Analise de DNA

O modulo de Cadastros e Analise de DNA é composto pela visualização de algumas informações a partir da sequência cadastrada no nosso banco de dados. Por exemplo, vamos cadastrar uma sequência de DNA qualquer para visualização.

Considere a seguinte sequência de DNA para ilustração:

>Teste DNA qualquer

AGCTTTTCATTCTGACTGCAACGGGCAATATGTCTCTGTGTGGATTAAAAAAGAGTGTCTGATAGCAGC

Esta é uma sequência no formato muito comum no estudo da bioinformática, é o formato “.fasta”, onde a primeira linha temos o cabeçalho da sequência:

>Teste DNA qualquer

E a partir da segunda linha temos a sequência do DNA:

AGCTTTTCATTCTGACTGCAACGGGCAATATGTCTCTGTGTGGATTAAAAAAGAGTGTCTGATAGCAGC

No ato do cadastro da sequência é preferível o cadastro pelo ambiente de administrador do Django, para isso acesse o endereço local: <http://127.0.0.1:8000/admin>

Django administration

Username:

admin

Password:

admin

Log in

Ao acessar o ambiente selecione a opção de “**Dnas**” dentro da coluna “**Modelagem**”

Django administration

Site administration

AUTHENTICATION AND AUTHORIZATION

Groups

+ Add

Change

Users

+ Add

Change

MODELAGEM

Dnas

+ Add

Change

Proteínas

+ Add

Change

PIPELINE

Arquivoss

+ Add

Change

Modeloss

+ Add

Change

Recent actions

My actions

+ DNA Correto

Dna

+ DNA Humano

Dna

+ Bactéria DNA

Dna

✖ 1º DNA Teste

Proteína

+ DNA Teste 1

Dna

+ 1º DNA Teste

Proteína

+ Beta-Glucosidase

Proteína

Agora clique em **ADD DNA +**

Select dna to change

ADD DNA +

Action: Go 0 of 5 selected

☐ DNA

☐ Sequencia Teste 2

☐ DNA Correto

☐ DNA Humano

☐ Bactéria DNA

☐ DNA Teste 1

5 dnas

E então teremos o acesso ao ambiente de cadastro de um novo DNA, cadastre da seguinte maneira:

The screenshot shows the 'Add dna' form in the Django administration interface. The form has three fields: 'Titulo:' with the value 'Sequencia Teste - DNA', 'Cabecalho:' with the value '>Teste DNA qualquer', and 'Sequencia:' with a long DNA sequence. At the bottom right, there are three buttons: 'Save and add another', 'Save and continue editing', and 'SAVE'. A red arrow points to the 'SAVE' button.

Obtendo a seguinte mensagem o DNA já foi cadastrado:

✓ The dna "Sequencia Teste - DNA" was added successfully.

Então podemos verificar suas informações no nosso Projeto dentro do Modulo de “Cadastro / Analise de DNA”.

The screenshot shows the 'Lista de DNA's Cadastrados' page. It has a dark background with a DNA double helix on the right. On the left, there is a list of DNA records: 'DNA Teste 1', 'Bactéria DNA', 'DNA Humano', 'DNA Correto', 'Sequencia Teste 2', and 'Sequencia Teste - DNA'. The 'Sequencia Teste - DNA' record is highlighted with a red box. Above the list is a green button labeled 'Adicionar DNA'.

Seremos redirecionados automaticamente para uma nova página onde constarão todas as informações da nossa sequência de DNA cadastrado:

The screenshot shows the 'Projeto Beta TCC' interface. It has a dark background with a DNA double helix on the right. At the top, it says 'ID: 7 - Proteína: Sequencia Teste - DNA'. Below this, there is a table with two columns: 'Sequência' and 'Converter'. The 'Sequência' column contains the DNA sequence 'AGCTTTTCATTCTGACTGCAACGGGCAATATGTCTCTGTGTGGATTAAAAAAGAGTGTCTGATAGCAGC'. The 'Converter' column has a button labeled 'Converter'. To the right of the table, there is a box with the text '20 - A', '21 - T', '17 - G', and '12 - C'. Below this, there is a box with the text '(A: 20, T: 21, G: 17, C: 12)'. At the bottom, there are four buttons: 'Fita Complementar do DNA inserido', 'Formar RNA mensageiro', 'Tabela de Códon', and 'Gerar a Proteína formada (aminoácidos)'. A 'Voltar' button is located at the top right.

Para efetuar o processo de Analise clique em “Converter”, neste caso serão apresentadas informações como:

Fita complementar do DNA.

RNA mensageiro .

Tabela de Códons (ilustração fixa).

Gerar a Proteína formada (sequência de aminoácidos a partir da Tabela de Códons)

Fita Complementar do DNA inserido

Fita:

TCGAAAAGTAAGACTGACGTTGCCGTTATACAGAGACACACCTAATTTTTTCTCACAGACTATCGTCG

Fita Completa:

ATGCCGTATATATACGATTATACGTAGCATCGTAGCCGATATCGGCGCGCCGATATTAATTAGCTACGTACGTAGCTAGCTAGCGCATTATAATATATATATATATGCATGCTAGCTACGTAGCATTAAATGCCGATGCCG

Sobreposição:

AGCTTTTCATTCTGACTGCAACGGGCAATATGCTCTGTGTGGATTAAAAAAGAGTGTCTCGAAAAGTAAGACTGACGTTGCCGTTATACAGAGACACACCTAATTTTTTCTCACAGTGATAGCAGC
ACTATCGTCG

Formar RNA mensageiro

RNA Mensageiro:

AGCUUUUCAAUUCUGACUGCAACGGGCAAUUGUCUCUGUGUGGAAUUAAAAAAGAGUGUCUGAUAGCAGC

Tabela de Códons					
1ª posição	2ª posição (meio)				3ª posição
	U	C	A	G	
U	Phe F	Ser S	Tyr Y	Cys C	U
	Phe F	Ser S	Tyr Y	Cys C	C
	Leu L	Ser S	stop	stop	A
	Leu L	Ser S	stop	Trp W	G
C	Leu L	Pro P	His H	Arg R	U
	Leu L	Pro P	His H	Arg R	C
	Leu L	Pro P	Gln Q	Arg R	A
	Leu L	Pro P	Gln Q	Arg R	G
A	Ile I	Thr T	Asn N	Ser S	U
	Ile I	Thr T	Asn N	Ser S	C
	Ile I	Thr T	Lys K	Arg R	A
	Met M	Thr T	Lys K	Arg R	G
G	Val V	Ala A	Asp D	Gly G	U
	Val V	Ala A	Asp D	Gly G	C
	Val V	Ala A	Glu E	Gly G	A
	Val V	Ala A	Glu E	Gly G	G

Gerar a Proteína formada (aminoácidos)
Arginina - Fenilalanina - Serina - Tirosina - Stop Códon - Leucina - Glutamina - Serina - Alanina - Isoleucina - Cisteína - Leucina - Cisteína - Valina - Aspartato - Lisina - Lisina - Serina - Valina - Isoleucina - Alanina -

Modulo comparar Sequências de Proteína

Este modulo é responsável por realizar um comparativo entre duas determinadas sequências de proteínas, oferecendo para nós como resultado, informações da quantidade de aparição de cada aminoacido na mesma e dados de alinhamento local e global.

Para a ilustração vamos usar duas sequências de proteínas:

>sp|P52407|E13B_HEVBR Glucan endo-1,3-beta-glucosidase, basic vacuolar isoform OS=Hevea brasiliensis OX=3981 GN=HGN1 PE=1 SV=2
MAISSSTSGTSSSFPSRTTVMLLFFFAASVGITDAQVGVCYGMQGNLPPVSEVIALYKKS
NITRMRIYDPNRAVLEALRGSNIELILGVPNSDLQSLTNPSNAKSWVQKNVRGFWSSVLFYR
IAVGNEISPVNRGTAWLAQFVLPAMRNIHDAIRSAGLQDQIKVSTAILDLTVGNSYPPSAGAF
RDDVRSYLDPIIGFLSSIRSPLLANIYPYFTYAYNPRDISLPYALFTSPSVVV
WDGQRGYKNLFDATLDALYSALERASGGSLVVVSESGWPSAGAFATFDNGRITYLSNLI
QHVKGGTPKRPNRAIETYLFAFMFDENKKQPEVEKHFGLFFPNKWQKYNLNFSAEKNWDIS
TEHNATILFLKSDM

e

>4hpg.pdb
QVGVCYGMQGNLPPVSEVIALYKKS NITRMRIYDPNQAVLEALRGSNIELILGVPNSDLQS
LTNPSNAKSWVQKNVRGFWSSVRFYIAVGNEISPVNRGTAWLAQFVLPAMRNIHDAIRSA
GLQDQIKVSTAILDLTVGNSYPPSAGAFRDDVRSYLNPIIRFLSSIRSPLLANIYPYFTYAGNP
RDISLPYALFTSPSVVVWDGQRGYKNLFDATLDALYSALERASGGSLVVVSESGWPSAGA
FAATFDNGRITYLSNLIQHVKRGTPKRPKRAIETYLFAFMFDENKKQPEVEKHFGLFFPNKWQ
KYNLNF

Insira as mesmas nos campos da seguinte forma:

Projeto Beta TCC	
Primeira Sequência	Segunda Sequência
Cabeçalho	>sp P52407 E13B_HEVBR Glucan endo-1,3-beta-glucosidase, basic vacuolar iso
Sequência	MAISSSTSGTSSSFPSRTTVMLLFFFAASVGITDAQVGVCYGMQGNLPPVSEVIALYKKS NITRMRIYDPNRAVLEALRGSNIELILGVPNSDLQSLTNPSNAKSWVQKNVRGFWSS VLFYIAVGNEISPVNRGTAWLAQFVLPAMRNIHDAIRSAGLQDQIKVSTAILDLTVGNS YPPSAGAFRDDVRSYLDPIIGFLSSIRSPLLANIYPYFTYAYNPRDISLPYALFTSPSVVV WDGQRGYKNLFDATLDALYSALERASGGSLVVVSESGWPSAGAFATFDNGRITYLSNLI QHVKGGTPKRPNRAIETYLFAFMFDENKKQPEVEKHFGLFFPNKWQKYNLNFSAEKNWDIS TEHNATILFLKSDM
Cabeçalho	>4hpg.pdb
Sequência	QVGVCYGMQGNLPPVSEVIALYKKS NITRMRIYDPNQAVLEALRGSNIELILGVPNSDLQS LTNPSNAKSWVQKNVRGFWSSVRFYIAVGNEISPVNRGTAWLAQFVLPAMRNIHDAIRSA GLQDQIKVSTAILDLTVGNSYPPSAGAFRDDVRSYLNPIIRFLSSIRSPLLANIYPYFTYAGNP RDISLPYALFTSPSVVVWDGQRGYKNLFDATLDALYSALERASGGSLVVVSESGWPSAGA FAATFDNGRITYLSNLIQHVKRGTPKRPKRAIETYLFAFMFDENKKQPEVEKHFGLFFPNKWQ KYNLNF
Alinhar	

Ao clicar em “Alinhar”, teremos um novo “Bloco” que nos da 3 botões para gerar os resultados, vamos clicar neles!

Resultado do Alinhamento	
Cabeçalho	>sp P52407 E13B_HEVBR Glucan endo-1,3-beta-glucosidase, basic vacuolar iso
Sequência	MAISSSTSGTSSSFPSRTTVMLLFFFAASVGITDAQVGVCYGMQGNLPPVSEVIALYKKS NITRMRIYDPNRAVLEALRGSNIELILGVPNSDLQSLTNPSNAKSWVQKNVRG
Cabeçalho	>4hpg.pdb
Sequência	QVGVCYGMQGNLPPVSEVIALYKKS NITRMRIYDPNQAVLEALRGSNIELILGVPNSDLQS LTNPSNAKSWVQKNVRGFWSSVRFYIAVGNEISPVNRGTAWLAQFVLPAM
Legenda: M: Match D: Diferente	
Detalhar Blastp Alinhamento	
Funções	

O botão de **Detalhar**, irá nos mostrar os detalhes individuais de cada sequência, voltado para a aparição de cada aminoacido dentro da mesma.

Alinhamento Detalhado das Sequências a partir do Back End Django | Comparação 1:

MAISSSTSGTSSSFPSRTTVMLLLFFFAASVGITDAQVGVCYGMQGNLPPVSEVIALYK
KSNITRMRIYDPNRAVLEALRGSNIELILGVPNSDLQSLTNPSNAKSWQKNVRGFWSSV
LFRYIavgNEISPVNrgTAWLAQFVLPAMrNIHDAIRsAGLQDQIKVSTaIDLTlVGNSY
PPSAGaFRDDVRSYLDPIIGfLSSIRsPllANIYPYfTYAYNPRDISLPYALFTSPSVVW
WDGQRgYKNLFDATLDALYSALERASGGsLEVWSESGWPSAGaFAATFDNGRTYLSNL
IQHVKGGTPKRPNRAIETYLfAMFDENKKQPEVEKHfGLFFPNKWQKYNLNFSAEKNWDI
STEHNATILFLKSDM

Quantidade de Aparições para cada Amionoacido
A: 32 C: 1 D: 17 E: 13 F: 21 G: 24 H: 4 I: 22 K: 15 L: 35 M: 7 N: 25 P: 21 Q: 11 R: 19 S: 40 T: 19
V: 25 W: 7 Y: 16

Porcentagem de Aparições para cada Amionoacido
A: 8.56% C: 0.27% D: 4.55% E: 3.48% F: 5.61% G: 6.42% H: 1.07% I: 5.88% K: 4.01% L: 9.36%
M: 1.87% N: 6.68% P: 5.61% Q: 2.94% R: 5.08% S: 10.7% T: 5.08% V: 6.68% W: 1.87% Y: 4.28%

Tamanho da Sequência: 374

Alinhamento Detalhado das Sequências a partir do Back End Django | Comparação 2:

QVGVCYGMQGNLPPVSEVIALYKSNITRMRIYDPNQAVALRGSNIELILGVPNSDL
QSLTNPSNAKSWQKNVRGFWSSVFRFYIAVGNEISPVNrgTAWLAQFVLPAMrNIHDAI
RSAGLQDQIKVSTaIDLTlVGNSYPPSAGaFRDDVRSYLNPIIRfLSSIRsPllANIYPY
FTYAGNPRDISLPYALFTSPSVVWWDGQRgYKNLFDATLDALYSALERASGGsLEVWSE
SGWPSAGaFAATFDNGRTYLSNLIQHVKRGTPKRPKRAIETYLfAMFDENKKQPEVEKHf
GLFFPNKWQKYNLNFSAEKNWDI

Quantidade de Aparições para cada Amionoacido
A: 26 C: 1 D: 13 E: 11 F: 16 G: 21 H: 3 I: 18 K: 14 L: 29 M: 4 N: 23 P: 20 Q: 12 R: 20 S: 29 T: 12
V: 23 W: 6 Y: 15

Porcentagem de Aparições para cada Amionoacido
A: 8.25% C: 0.32% D: 4.13% E: 3.49% F: 5.08% G: 6.67% H: 0.95% I: 5.71% K: 4.44% L: 9.21%
M: 1.27% N: 7.3% P: 6.35% Q: 3.81% R: 6.35% S: 9.21% T: 3.81% V: 7.3% W: 1.9% Y: 4.76%

Tamanho da Sequência: 315

O botão de **alinhamento**, irá nos mostrar o alinhamento das duas sequências, evidenciando com o caracter “+” onde não houve um “**match**”, ou simplificando, onde as posições de cada sequência não coincidiram, este exemplo também pode ser visto na imagem acima, onde os caracteres evidenciados em **vermelho** representam a ocorrência de “**Miss Match**” e os verdes a ocorrência de um “**Match**”. Esta função caracteriza o que chamamos de **Alinhamento local**, que é comparar caracter por caracter em busca de “**Matches**” .

Simulação:

MAISSSTSGTSSSFPSRTTVMLLLFFFAASVGITDAQVGVCYGMQGNLPPVSEVIALYK
++++++P++++L++++I+D+++V+++G+N++++V++++
QVGVCYGMQGNLPPVSEVIALYKSNITRMRIYDPNQAVALRGSNIELILGVPNSDL

KSNITRMRIYDPNRAVLEALRGSNIELILGVPNSDLQSLTNPSNAKSWQKNVRGFWSSV
+S++++S++++V+N++++V+++R++++
QSLTNPSNAKSWQKNVRGFWSSVFRFYIAVGNEISPVNrgTAWLAQFVLPAMrNIHDAI

LFRYIavgNEISPVNrgTAWLAQFVLPAMrNIHDAIRsAGLQDQIKVSTaIDLTlVGNSY
++++S++++T++++A++++D++++S++++L++++Y
RSAGLQDQIKVSTaIDLTlVGNSYPPSAGaFRDDVRSYLNPIIRfLSSIRsPllANIYPY

PPSAGaFRDDVRSYLDPIIGfLSSIRsPllANIYPYfTYAYNPRDISLPYALFTSPSVVW
+++AG++RD+++Y++++T++++V++
FTYAGNPRDISLPYALFTSPSVVWWDGQRgYKNLFDATLDALYSALERASGGsLEVWSE

WDGQRgYKNLFDATLDALYSALERASGGsLEVWSESGWPSAGaFAATFDNGRTYLSNL
++++G++++S+L++++A++++
SGWPSAGaFAATFDNGRTYLSNLIQHVKRGTPKRPKRAIETYLfAMFDENKKQPEVEKHf

IQHVKGGTPKRPNRAIETYLfAMFDENKKQPEVEKHfGLFFPNKWQKYNLNFSAEKNWDI
++++K++++
GLFFPNKWQKYNLNFSAEKNWDI

STEHNATILFLKSDM

Obs: O resultado é dividido em 3 linhas, sendo a primeira a 1º sequência inserida, a segunda o comparativo e a terceira a 2º sequência inserida. Cada linha possui exatamente **60** caracteres.

O botão **blastp**, utiliza uma ferramenta do BioPython para realizar o **Alinhamento Global**, buscando encontrar na sequência como um todo as “Cadeias de caracteres” semelhantes, agora não mais utilizando o alinhamento local (comparando 1 a 1), mas sim um algoritmo capaz de encontrar semelhanças em cadeias.

```
Blastp:
QVGVCYGMQGNLPPVSEVIALYKKSNIITRMRIYDPNRAVLEALRGSNIELILGVPNSDL
QVGVCYGMQGNLPPVSEVIALYKKSNIITRMRIYDPN+AVLEALRGSNIELILGVPNSDL
QVGVCYGMQGNLPPVSEVIALYKKSNIITRMRIYDPNQAVLEALRGSNIELILGVPNSDL

QSLTNPSNAKSNWQKNVRGFWSSVLFRIYAVGNEISPVNRGTAWLAQFVLPAMRNIHDAI
QSLTNPSNAKSNWQKNVRGFWSSV FRIYAVGNEISPVNRGTAWLAQFVLPAMRNIHDAI
QSLTNPSNAKSNWQKNVRGFWSSVRFRIYAVGNEISPVNRGTAWLAQFVLPAMRNIHDAI

RSAGLQDQIKVSTAILTLVGNSYPPSAGAFRDDVRSYLDPIIGFLSSIRSPLLANIYPY
RSAGLQDQIKVSTAILTLVGNSYPPSAGAFRDDVRSYL+PII FLSSIRSPLLANIYPY
RSAGLQDQIKVSTAILTLVGNSYPPSAGAFRDDVRSYLNPIIRFLSSIRSPLLANIYPY

FTYAYNPRDISLPYALFTSPSVVWVDGQRGYKNLFDATLDALYSALERASGGGSLEVWSE
FTYA NPRDISLPYALFTSPSVVWVDGQRGYKNLFDATLDALYSALERASGGGSLEVWSE
FTYAGNPRDISLPYALFTSPSVVWVDGQRGYKNLFDATLDALYSALERASGGGSLEVWSE

SGWPSAGAFATFDNGRTYLSNLIQHVKGGTPKRPNRAIETYLFAFMDENKKQPEVEKHF
SGWPSAGAFATFDNGRTYLSNLIQHVK GTPKRP RAIETYLFAFMDENKKQPEVEKHF
SGWPSAGAFATFDNGRTYLSNLIQHVKRGTPKRPKRAIETYLFAFMDENKKQPEVEKHF

GLFFPNKWQKYNLNF
GLFFPNKWQKYNLNF
GLFFPNKWQKYNLNF
```

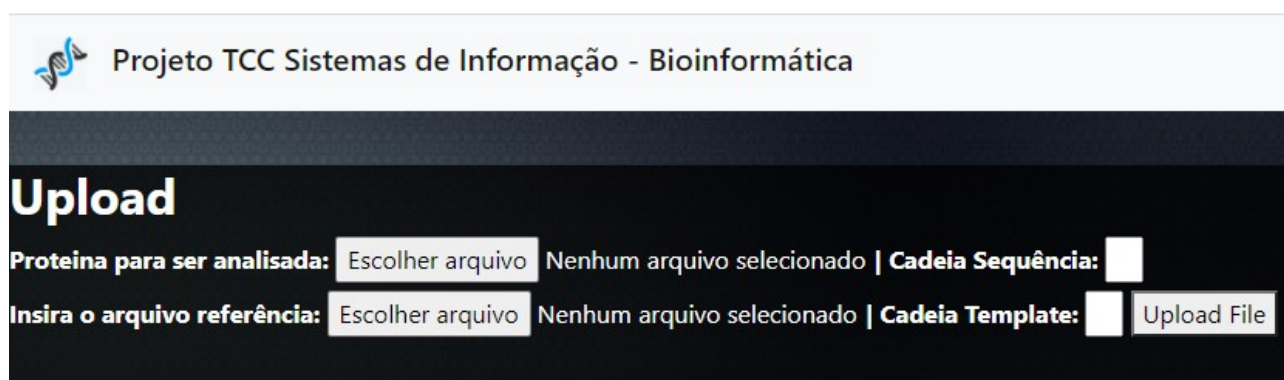
Obs: O resultado é dividido em 3 linhas, sendo a primeira a 1º sequência inserida, a segunda o comparativo e a terceira a 2º sequência inserida. Cada linha possui exatamente 60 caracteres. Note que o resultado é menor, pois o blastp neste caso, evidência somente os trechos da sequência em que aconteceram de fato o “Alinhamento Global”, o fim e o início de cada sequência do Alinhamento Global é destacado por um “sape” de um espaço vazio.

Em **Branco** temos os trechos das duas sequências do **alinhamento global** destacadas pelo **Blastp**. Em **Verde** temos os **resultados** do **alinhamento global** em **comparativo** uma com a outra.

Módulo de Modelagem 3D

Este módulo é responsável por criar um modelo de Proteína 3D a partir da estrutura de um Template referência. Este processo é denominado como “modelagem por homologia”, o processo de modelagem 3d requer a utilização de vários softwares da bioinformática, em geral, via linha de comando (Terminal). Porém, neste projeto foram separadas as funções de cada software, para funcionarem de forma “Conectada”, ou melhor dizendo, seguindo uma ordem de execução (pipeline), onde os comandos serão chamados em sequência e seus arquivos também gerados em sequência.

Para executar esta rotina é simples, primeiro acesse o módulo responsável por gerar os modelos 3d









The screenshot shows a web interface titled "Projeto TCC Sistemas de Informação - Bioinformática". Below the title is a dark blue header with the word "Upload" in white. The main area has a black background with white text. It contains two rows of input fields. The first row is labeled "Proteína para ser analisada:" and has a button "Escolher arquivo", the text "Nenhum arquivo selecionado", a label "Cadeia Sequência:" followed by a text input field. The second row is labeled "Insira o arquivo referência:" and has a button "Escolher arquivo", the text "Nenhum arquivo selecionado", a label "Cadeia Template:" followed by a text input field, and a button "Upload File".

Em seguida selecione os arquivos para a execução da rotina onde:

“Proteína para ser analisada”, será um arquivo no formato tipo “.fasta”, no caso a sequência que iremos gerar o modelo 3D.

“Insira o arquivo referência”, neste campo vamos inserir um arquivo “Template” no formato “.pdb”.

Para facilitar, dentro do projeto já temos os arquivos separados da proteína “Beta-Glucosidase”, dentro do diretório “TCC/BetaTCC/media/**modelos**/”.

 4hpg.pdb	First commit	22 days ago
 4hpg.pdb.fasta	First commit	22 days ago
 Alinhamento Detalhado das Sequências a partir do B	First commit	22 days ago
 beta_glucosidade10.fasta	First commit	22 days ago
 beta_glucosidase.fasta	First commit	22 days ago
 beta_glucosidase1.fasta	First commit	22 days ago

Então teremos:

“Proteína para ser analisada”: beta_glucosidase10.fasta

“Insira o arquivo referência”: 4hpg.pdb

Ficando desta forma:

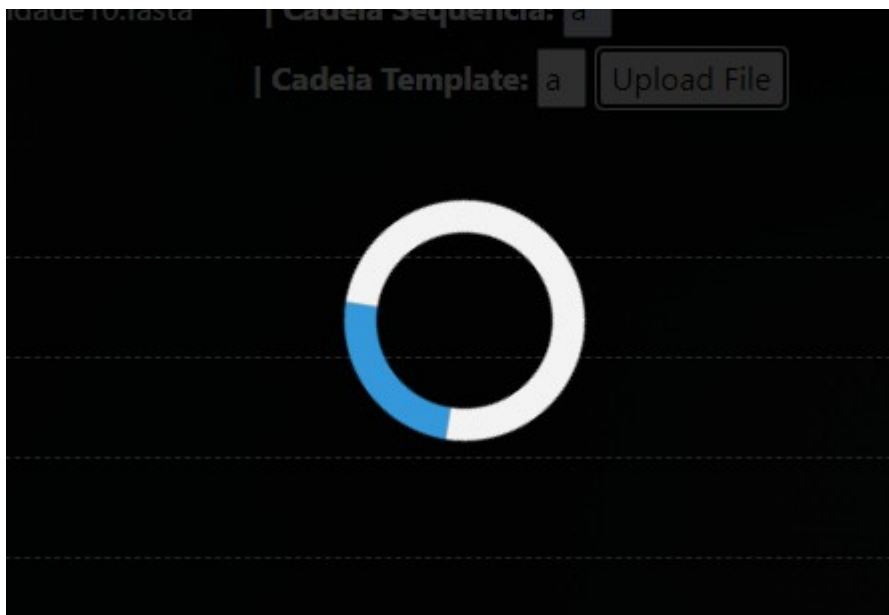
Projeto TCC Sistemas de Informação - Bioinformática

Upload

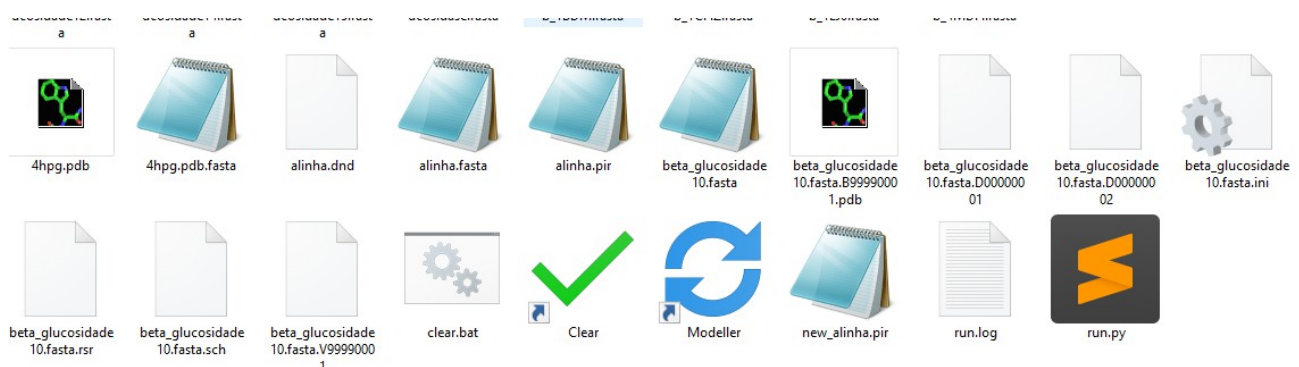
Proteína para ser analisada: Escolher arquivo beta_glucosidade10.fasta | Cadeia Sequência: a

Insira o arquivo referência: Escolher arquivo 4hpg.pdb | Cadeia Template: a Upload File

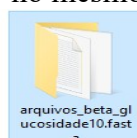
Em “Cadeia Sequência” e “Cadeia Template”, teremos que colocar a cadeia que será utilizada, neste caso utilizaremos a Cadeia “A”. Então clique em “Upload File”.



O processo de modelagem 3D começara a gerar arquivos, você pode acompanhar eles sendo criados dentro do diretório “media”



Após a finalização o Script irá criar automaticamente para você um novo “Diretório” e moverá todos os arquivos importantes para o modelo no mesmo.



Nome	Data de modificação	Tipo	Tamanho
4hpg.pdb	17/11/2020 19:58	Protein Data Bank ...	866 KB
4hpg.pdb.fasta	17/11/2020 19:58	Arquivo FASTA	1 KB
alinha.fasta	17/11/2020 19:58	Arquivo FASTA	1 KB
alinha.pir	17/11/2020 19:58	Arquivo PIR	1 KB
beta_glucosidade10.fasta	17/11/2020 19:58	Arquivo FASTA	1 KB
beta_glucosidade10.fasta.B99990001.pdb	17/11/2020 19:59	Protein Data Bank ...	227 KB
beta_glucosidade10.fasta.B99990002.pdb	17/11/2020 19:59	Protein Data Bank ...	227 KB
beta_glucosidade10.fasta.B99990003.pdb	17/11/2020 19:59	Protein Data Bank ...	227 KB
new_alinha.pir	17/11/2020 19:58	Arquivo PIR	1 KB
run.log	17/11/2020 19:59	Documento de Te...	84 KB

Por padrão o script está programado para gerar somente “2 modelos 3D”, em virtude de ser um processo um pouco demorado, no entanto a quantidade de modelos gerados por padrão pode ser ajustada no arquivo “views.py” no diretório: “/pipeline”

__pycache__	16/11/2020 20:33	Pasta de arquivos	
migrations	19/10/2020 16:35	Pasta de arquivos	
templates	13/07/2020 16:38	Pasta de arquivos	
__init__.py	13/07/2020 16:27	Arquivo PY	0 KB
admin.py	19/10/2020 17:50	Arquivo PY	1 KB
apps.py	13/07/2020 16:27	Arquivo PY	1 KB
forms.py	20/10/2020 15:51	Arquivo PY	1 KB
models.py	19/10/2020 17:08	Arquivo PY	1 KB
tests.py	13/07/2020 16:27	Arquivo PY	1 KB
urls.py	26/10/2020 12:44	Arquivo PY	1 KB
views.py	16/11/2020 20:33	Arquivo PY	17 KB

Na função "criaScript", procure pela variável “script” que está recebendo um valor do tipo “string” referente ao script gerado de forma dinâmica. No final desta linha, temos o seguinte conteúdo:

```
''+template.name+'')\na.starting_model = 1\na.ending_model = 2\na.make() \n"
```

onde:

“a.starting_model = 1” é a configuração inicial do modelo de “Start”, não mexeremos aqui.

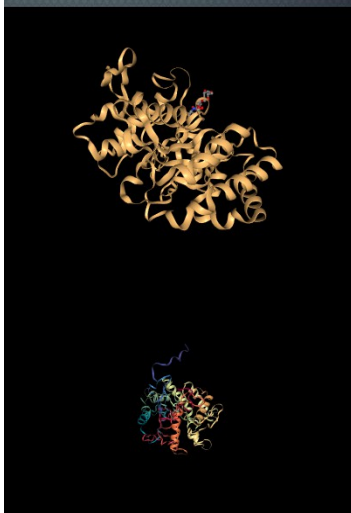
“a.ending_model = 2” é aqui onde podemos alterar a quantidade de modelos 3d gerados alterando o “2” pela quantidade desejada.

Voltando a execução do módulo, após o carregamento desta rotina, teremos o seguinte resultado:

- **beta_glucosidade10.fasta**

Ao clicar seremos redirecionados a uma nova janela.

Carregado com Sucesso!Voltar



Modelo: 4hpg.pdb Score: **Template - Sem Score**
HEADER HYDROLASE, ALLERGEN 23-OCT-12 4HPG
TITLE CRYSTAL STRUCTURE OF A GLYCOSYLATED BETA-1,3-GLUCANASE (HEV B 2), AN
TITLE 2 ALLERGEN FROM HEVEA BRASILIENSIS
COMPND MOL_ID: 1;
COMPND 2 MOLECULE: BETA-1,3-GLUCANASE;
COMPND 3 CHAIN: A, B, C, D;
COMPND 4 EC: 3.2.1.39
Bloquear Scroll Download Arquivo

Modelo: beta_glucosidade10.fasta.B99990001.pdb Score: **2072.6064**
EXPDTA THEORETICAL MODEL, MODELLER 9.24 2020/11/17 19:59:12
REMARK 6 MODELLER OBJECTIVE FUNCTION: 2072.6064
REMARK 6 MODELLER BEST TEMPLATE % SEQ ID: 97.778
REMARK 6 SEQUENCE: beta_glucosidade10.fasta
REMARK 6 ALIGNMENT: new_alinha.pir
REMARK 6 SCRIPT: run.py
REMARK 6 TEMPLATE: 4hpg.pdb 2:A - 316:A MODELS 38: - 352: AT 97.8%
Bloquear Scroll Download Arquivo

Aqui poderemos verificar a proteína 3D, interagir com ela, mover, visualizar, baixar o modelo e visualizar os seus dados.

Finalizando...

Dadas configurações o sistema já está totalmente configurado e pronto para ser utilizado em seu computador.

Deixamos este tópico final para agradecer a todos os envolvidos no desenvolvimento deste projeto, em especial ao professor Camilo César Perucci que nos deu a possibilidade de trabalhar com uma área fora da atuação da nossa grade e a instituição que nestes 4 anos de curso, sempre manteve com a prestação de excelentes aulas e conteúdo para nossa formação.

Aos demais futuros alunos que estiverem lendo este documento, fiquem com o nosso agradecimento, a jornada pode ser difícil, mas o reconhecimento do seu esforço e aprendizado sempre virá à tona.

Para qualquer dúvida ou informação extra necessária a este documento, por favor não exite em nos contatar, deixarei nossos E-mails para contato ao final.

Nosso agradimento a todos.

Atenciosamente,

Rafael Franco Nascimento.

E-mail Institucional: rafael_nascimen@alunos.fho.edu.br.

E-mail Pessoal: rafaelfranco.gu@gmail.com.

Rodrigo Santiago Claro Filho.

E-mail Institucional: 33rodrigossantiago@alunos.fho.edu.br.

E-mail Pessoal: 33rodrigossantiago@gmail.com.