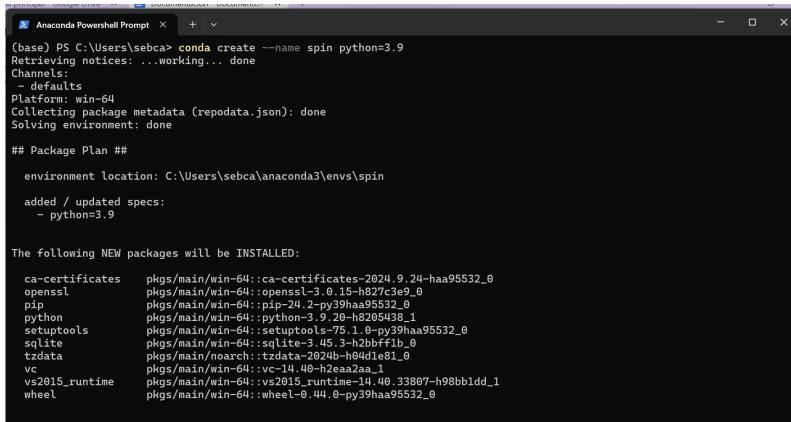


Documentación Conexión QC

- Instalar Anaconda (Si es en Windows, ejecutar todos los comando en Anaconda Powershell Prompt)
- Crear nuevo entorno de trabajo
conda create - -name nombre_entorno python=3.9



```
(base) PS C:\Users\sebca> conda create --name spin python=3.9
Retrieving notices: ...working... done
Channels:
- defaults
Platform: win-64
Collecting package metadata (repodata.json): done
Solving environment: done

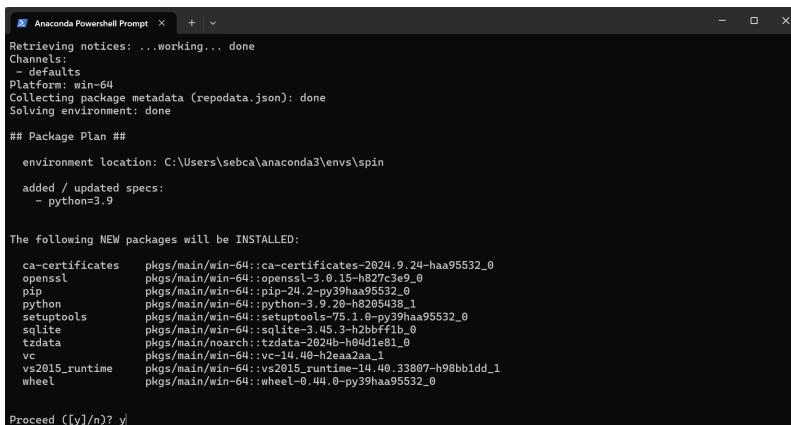
## Package Plan ##

environment location: C:\Users\sebca\anaconda3\envs\spin

added / updated specs:
- python=3.9

The following NEW packages will be INSTALLED:

ca-certificates      pkgs/main/win-64::ca-certificates-2024.9.24-haa95532_0
openssl              pkgs/main/win-64::openssl-3.0.15-h827c3e9_0
pip                  pkgs/main/win-64::pip-24.2-py39haa95532_0
python               pkgs/main/win-64::python-3.9.20-h8205438_1
setuptools           pkgs/main/win-64::setuptools-75.1.0-py39haa95532_0
sqlite               pkgs/main/win-64::sqlite-3.45.3-h2bbff1b_0
tzdata               pkgs/main/noarch::tzdata-2024b-h04d1e81_0
vc                   pkgs/main/win-64::vc-14.40-h2ea2aa_1
vs2015_runtime       pkgs/main/win-64::vs2015_runtime-14.40.33807-h98bb1dd_1
wheel                pkgs/main/win-64::wheel-0.44.0-py39haa95532_0
```



```
Retrieving notices: ...working... done
Channels:
- defaults
Platform: win-64
Collecting package metadata (repodata.json): done
Solving environment: done

## Package Plan ##

environment location: C:\Users\sebca\anaconda3\envs\spin

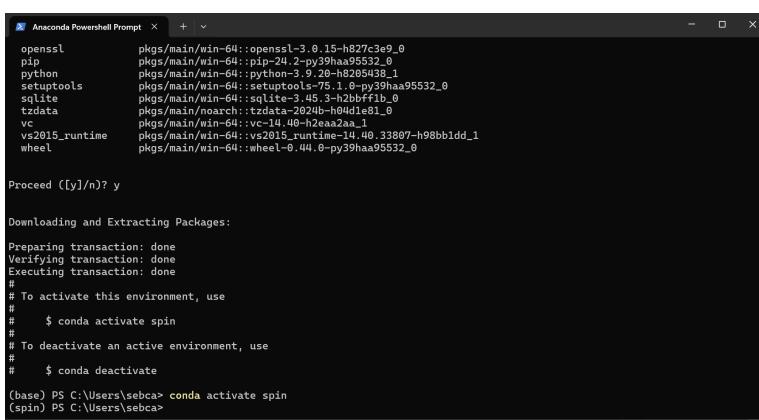
added / updated specs:
- python=3.9

The following NEW packages will be INSTALLED:

ca-certificates      pkgs/main/win-64::ca-certificates-2024.9.24-haa95532_0
openssl              pkgs/main/win-64::openssl-3.0.15-h827c3e9_0
pip                  pkgs/main/win-64::pip-24.2-py39haa95532_0
python               pkgs/main/win-64::python-3.9.20-h8205438_1
setuptools           pkgs/main/win-64::setuptools-75.1.0-py39haa95532_0
sqlite               pkgs/main/win-64::sqlite-3.45.3-h2bbff1b_0
tzdata               pkgs/main/noarch::tzdata-2024b-h04d1e81_0
vc                   pkgs/main/win-64::vc-14.40-h2ea2aa_1
vs2015_runtime       pkgs/main/win-64::vs2015_runtime-14.40.33807-h98bb1dd_1
wheel                pkgs/main/win-64::wheel-0.44.0-py39haa95532_0

Proceed ([y]/n)? y

Downloading and Extracting Packages:
```



```
openssl              pkgs/main/win-64::openssl-3.0.15-h827c3e9_0
pip                  pkgs/main/win-64::pip-24.2-py39haa95532_0
python               pkgs/main/win-64::python-3.9.20-h8205438_1
setuptools           pkgs/main/win-64::setuptools-75.1.0-py39haa95532_0
sqlite               pkgs/main/win-64::sqlite-3.45.3-h2bbff1b_0
tzdata               pkgs/main/noarch::tzdata-2024b-h04d1e81_0
vc                   pkgs/main/win-64::vc-14.40-h2ea2aa_1
vs2015_runtime       pkgs/main/win-64::vs2015_runtime-14.40.33807-h98bb1dd_1
wheel                pkgs/main/win-64::wheel-0.44.0-py39haa95532_0

Proceed ([y]/n)? y

Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
# $ conda activate spin
#
# To deactivate an active environment, use
# $ conda deactivate

(base) PS C:\Users\sebca> conda activate spin
(spin) PS C:\Users\sebca>
```

- Instalar bibliotecas necesarias

```
pip install spinqit
```

```
pip install spinqit numpy==1.21.0
```

The image contains three vertically stacked screenshots of an Anaconda PowerShell Prompt window. Each screenshot shows the command being run and the resulting output of the pip install command.

```
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
# $ conda activate spin
#
# To deactivate an active environment, use
#
# $ conda deactivate
(base) PS C:\Users\sebca> conda activate spin
(spin) PS C:\Users\sebca> pip install spinqit
Collecting spinqit
  Using cached spinqit-0.2.3-cp39-cp39-win_amd64.whl.metadata (881 bytes)
Collecting numpy (from spinqit)
  Using cached numpy-1.21.0-cp39-cp39-win_amd64.whl.metadata (59 kB)
Collecting scipy (from spinqit)
  Using cached scipy-1.10.1-cp39-cp39-win_amd64.whl.metadata (60 kB)
Collecting scikit-learn (from spinqit)
  Using cached scikit_learn-0.21.3-cp39-cp39-win_amd64.whl.metadata (13 kB)
Collecting torch (from spinqit)
  Using cached torch-1.8.0-cp39-cp39-win_amd64.whl.metadata (27 kB)
Collecting autograd=1.8.0 (from spinqit)
  Using cached autograd-1.8.0-py3-none-any.whl.metadata (704 bytes)
Collecting psutil (from spinqit)
  Using cached psutil-3.7.0-cp39-cp39-win_amd64.whl.metadata (22 kB)
```



```
Using cached contourpy-1.3.0-cp39-cp39-win_amd64.whl (211 kB)
Using cached cypher-0.12.1-py3-none-any.whl (8.3 kB)
Using cached fonttools-4.24.0-cp39-cp39-win_amd64.whl (2.2 MB)
Using cached future-1.16.0-py3-none-any.whl (491 kB)
Using cached idna-3.1-py3-none-any.whl (70 kB)
Using cached importlib_resources-6.4.5-py3-none-any.whl (36 kB)
Using cached joblib-1.4.2-py3-none-any.whl (301 kB)
Using cached kiwisolver-1.7.0-cp39-cp39-win_amd64.whl (55 kB)
Using cached libtiff-4.1.0-py3-none-any.whl (51 kB)
Using cached packaging-21.3-py3-none-any.whl (53 kB)
Using cached pillow-8.2.0-cp39-cp39-win_amd64.whl (2.6 MB)
Downloading pynarsing-3.2.0-py3-none-any.whl (106 kB)
Using cached python_dateutil-2.9.0-py0-py2.py3-none-any.whl (229 kB)
Using cached threadedpoolctl-3.5.0-py3-none-any.whl (18 kB)
Using cached typing_extensions-4.12.2-py3-none-any.whl (37 kB)
Using cached urllib3-2.2.0-py3-none-any.whl (126 kB)
Using cached filelock-3.1.1-py3-none-any.whl (16 kB)
Using cached fsspec-2024.9.0-py3-none-any.whl (179 kB)
Using cached jinja2-3.1.4-py3-none-any.whl (133 kB)
Using cached networkx-3.2-py3-none-any.whl (1.6 MB)
Using cached nose-safefile-3.0.0-cp39-cp39-win_amd64.whl (15 kB)
Using cached six-1.16.0-py2.py3-none-any.whl (11 kB)
Using cached texttable-1.7.0-py2.py3-none-any.whl (10 kB)
Using cached zipp-3.20.2-py3-none-any.whl (9.2 kB)
Installing collected packages: texttable, python-constraint, mpmath, antlr4-python3-runtime, zipp, urllib3, typing-extensions, threadpoolctl, sympy, six, pyParsing, pycryptodome, pybind11, psutil, pillow, packaging, numpy, networkx, MarkupSafe, kiwisolver, joblib, iGraph, idna, future, fsspec, fonttools, filelock, cypher, charset-normalizer, certifi, autoray, scipy, rustworkx, requests, python-igraph, python-dateutil, jinja2, importlib-resources, contourpy, autograd, torch, scikit-learn, networkx, noisyopt, matplotlib, spinqit
```



```
(spin) PS C:\Users\sebca> pip install spinqit numpy==1.21.0
Requirement already satisfied: spinqit in c:\users\sebca\anaconda3\envs\spin\lib\site-packages (0.2.3)
Requirement already satisfied: numpy==1.21.0 in c:\users\sebca\anaconda3\envs\spin\lib\site-packages (1.21.0)
Requirement already satisfied: scipy in c:\users\sebca\anaconda3\envs\spin\lib\site-packages (from spinqit) (1.13.1)
Requirement already satisfied: scikit-learn in c:\users\sebca\anaconda3\envs\spin\lib\site-packages (from spinqit) (1.5.2)
Requirement already satisfied: torch in c:\users\sebca\anaconda3\envs\spin\lib\site-packages (from spinqit) (2.4.1)
Requirement already satisfied: autograd==1.5.0 in c:\users\sebca\anaconda3\envs\spin\lib\site-packages (from spinqit) (1.5)
Requirement already satisfied: psutil in c:\users\sebca\anaconda3\envs\spin\lib\site-packages (from spinqit) (6.0.0)
Requirement already satisfied: networkx in c:\users\sebca\anaconda3\envs\spin\lib\site-packages (from spinqit) (2.15.1)
Requirement already satisfied: python-igraph==0.9.10 in c:\users\sebca\anaconda3\envs\spin\lib\site-packages (from spinqit) (0.9.10)
Requirement already satisfied: requests in c:\users\sebca\anaconda3\envs\spin\lib\site-packages (from spinqit) (2.32.3)
Requirement already satisfied: antlr4-python3-runtime==4.9.2 in c:\users\sebca\anaconda3\envs\spin\lib\site-packages (from spinqit) (4.9.2)
Requirement already satisfied: python-constraint in c:\users\sebca\anaconda3\envs\spin\lib\site-packages (from spinqit) (1.4.0)
Requirement already satisfied: numpy>=1.17.0 in c:\users\sebca\anaconda3\envs\spin\lib\site-packages (from spinqit) (1.17.0)
Requirement already satisfied: pycryptodome==3.11.0 in c:\users\sebca\anaconda3\envs\spin\lib\site-packages (from spinqit) (3.11.0)
Requirement already satisfied: autoray==0.6.1 in c:\users\sebca\anaconda3\envs\spin\lib\site-packages (from spinqit) (0.6.1)
Requirement already satisfied: noisyopt==0.2.2 in c:\users\sebca\anaconda3\envs\spin\lib\site-packages (from spinqit) (0.2.2)
Requirement already satisfied: future>=0.15.2 in c:\users\sebca\anaconda3\envs\spin\lib\site-packages (from autograd==1.5.0->spinqit) (0.15.2)
```

- Crear programa en python o jupyter notebook el cual utilice el entorno creado anteriormente
- Código de ejemplo:

```
from spinqit import get_nmr, get_compiler, Circuit, NMRConfig
from spinqit import H, CX, Rx
```

```

engine = get_nmr()
comp = get_compiler("native")

circ = Circuit()

q = circ.allocateQubits(2)
circ << (H, q[0])
circ << (H, q[1])


exe = comp.compile(circ, 0)
config = NMRCConfig()
config.configure_shots(1024)
config.configure_ip("172.27.52.229")
config.configure_port(8989)
config.configure_account("SpinQ001","123456") #usuario y
contraseña
config.configure_task("Test-JUT","Test-JUT") #nombre de la tarea

result = engine.execute(exe, config)
print(result.probabilities)

```

Explicación del código:

- Se importan funciones y clases de la librería **spinqit** para crear y ejecutar circuitos cuánticos en una simulación basada en NMR (resonancia magnética nuclear):

```

from spinqit import get_nmr, get_compiler, Circuit, NMRCConfig
from spinqit import H, CX, RX

```

- Configuración del motor de simulación y compilador:

```

engine = get_nmr()
comp = get_compiler("native")

```

- Creación del circuito:

```

circ = Circuit()
q = circ.allocateQubits(2) #Debido a que el computador cuántico
trabaja con dos qubits reales, solo podemos agregar esta cantidad
al trabajar con get_nmr
circ << (H, q[0])

```

```
circ << (H, q[1])
```

- Compilación del circuito:

```
exe = comp.compile(circ, 0)
```

- Configuraciones necesarias para conectarse al computador cuántico:

```
config = NMRConfig()
config.configure_shots(1024)
config.configure_ip("172.27.52.229") #IP del Computador cuantico
config.configure_port(8989) #puerto al cual escucha el computador
cuantico
config.configure_account("SpinQ001","123456")           #usuario      y
contraseña
config.configure_task("Test-JUT","Test-JUT") #tarea
```

- Ejecución y resultados:

```
result = engine.execute(exe, config)
print(result.probabilities)
```

En caso de haber creado archivo .py, ejecutar el comando **python .nombre_app.py**

The screenshot shows a Windows command-line interface window titled "Anaconda Powershell Prompt". It displays two separate sessions of the PowerShell command-line.

The first session shows the directory structure and files in the "Proyecto" folder:

Mode	LastWriteTime	Length	Name
-a---l	07-10-2024	21:01	Bootcamp Lab0 Hola mundo
-a---l	03-09-2024	15:48	Ejercicio Computacion
-a---l	09-10-2024	15:08	Proyecto
-a---l	06-09-2024	19:50	20573 Ejercicio2.ipynb
-a---l	23-09-2024	12:26	15741 Proyecto Computación Cuántica.docx

The second session shows the execution of a Python script named "prueba.py":

```
(spin) PS C:\Users\sebca\OneDrive\Documentos\Computación Cuántica> cd .\Proyecto\
(spin) PS C:\Users\sebca\OneDrive\Documentos\Computación Cuántica\Proyecto> ls

Directorio: C:\Users\sebca\OneDrive\Documentos\Computación Cuántica\Proyecto

Mode                LastWriteTime         Length Name
----                -----        ----
d----l    09-10-2024     15:03             Prot2
d----l    09-10-2024     14:07             Prototipo
-a---l    09-10-2024     10:32          1655 AlgoritmoPrueba.py
-a---l    09-10-2024     13:04          8608 AlgoritmoPrueba.ipynb
-a---l    14-10-2024     23:09            714 prueba.py
-a---l    11-10-2024     14:01          1741 Test.ipynb
-a---l    08-10-2024     16:30            233 x.txt

(spin) PS C:\Users\sebca\OneDrive\Documentos\Computación Cuántica\Proyecto> python .\prueba.py
```

Esto fue probado tanto en el sistema operativo de Windows como en Ubuntu