



۸۱۰۱۰۱۵۵۸

پردازش اطلاعات کوانتومی  
نام و نام خانوادگی: مهدی وجهی



پروژه ۱

## ۱ آماده‌سازی محیط و اجرای پروژه

برای اجرای شبیه‌سازی مدار کوانتومی، ابتدا نیاز به آماده‌سازی یک محیط ایزوله پایتون بود تا از تداخل بسته‌های نرم‌افزاری جلوگیری شود. سپس بسته‌های تخصصی مورد نیاز نصب و در نهایت کد پروژه در یک نوت‌بوک جویپتر اجرا گردید.

### ۱.۱ ایجاد و فعال‌سازی محیط مجازی

به منظور مدیریت وابستگی‌های (*dependencies*) پروژه، یک محیط مجازی پایتون (*Virtual Environment*) با استفاده از ماژول `venv` در پوشه‌ی `env/` ایجاد شد.

ابتدا به دایرکتوری مورد نظر رفته و سپس محیط مجازی ایجاد شد:

```
env cd  
.venv venv -m python
```

پس از ایجاد، این محیط با دستور زیر فعال گردید تا تمامی بسته‌های بعدی در این محیط ایزوله نصب شوند:

```
.venv/bin/activate source
```

### ۲.۱ نصب بسته‌های مورد نیاز

پس از فعال‌سازی محیط، بسته‌های اصلی با استفاده از `pip` نصب شدند.

۱. نصب *Qiskit*: بسته اصلی `qiskit` که کتابخانه محاسبات کوانتومی است، نصب شد.

۲. نصب کرنل *Jupyter*: برای اینکه بتوانیم از این محیط مجازی در جویپتر نوت‌بوک استفاده کنیم، بسته

`ipykernel` نصب گردید.

۳. نصب بسته‌های ترسیم (*Visualization*): همانطور که در کد پروژه مشخص است (استفاده از `(circ.draw('mpl'))`،

برای ترسیم مدار به کتابخانه `matplotlib` نیاز است. همچنین بسته `pylatexenc` برای رندر صحیح عبارات

ریاضی و نمادها در ترسیم‌های `matplotlib` نصب شد.

خلاصه دستورات نصب موفق در ترمینال به شرح زیر است:

```
qiskit install pip
ipykernel install pip
matplotlib install pip
pylatexenc install pip
```

### ۳.۱ ثبت کرنل (Kernel) برای جوپیتر

پس از نصب ipykernel، محیط مجازی فعال باید به عنوان یک "کرنل" به جوپیتر معرفی می‌شد تا در لیست کرنل‌های قابل انتخاب در جوپیتر نوت‌بوک نمایش داده شود. این کار با دستور زیر و با استفاده از فلگ --user (به دلیل عدم نیاز به دسترسی root و رفع خطای *Permission Denied*) انجام شد و کرنلی با نام qipCourse ثبت گردید:

```
-user- -name=qipCourse- install ipykernel -m python
```

### ۴.۱ اجرا و دریافت خروجی

در نهایت، جوپیتر نوت‌بوک راه‌اندازی شد و نوت‌بوک جدیدی با استفاده از کرنل qipCourse ایجاد گردید. کد پروژه در سلول نوت‌بوک اجرا شد.

این کد یک مدار کوانتومی با ۳ کیوبیت (QuantumCircuit(3)) ایجاد می‌کند. سپس:

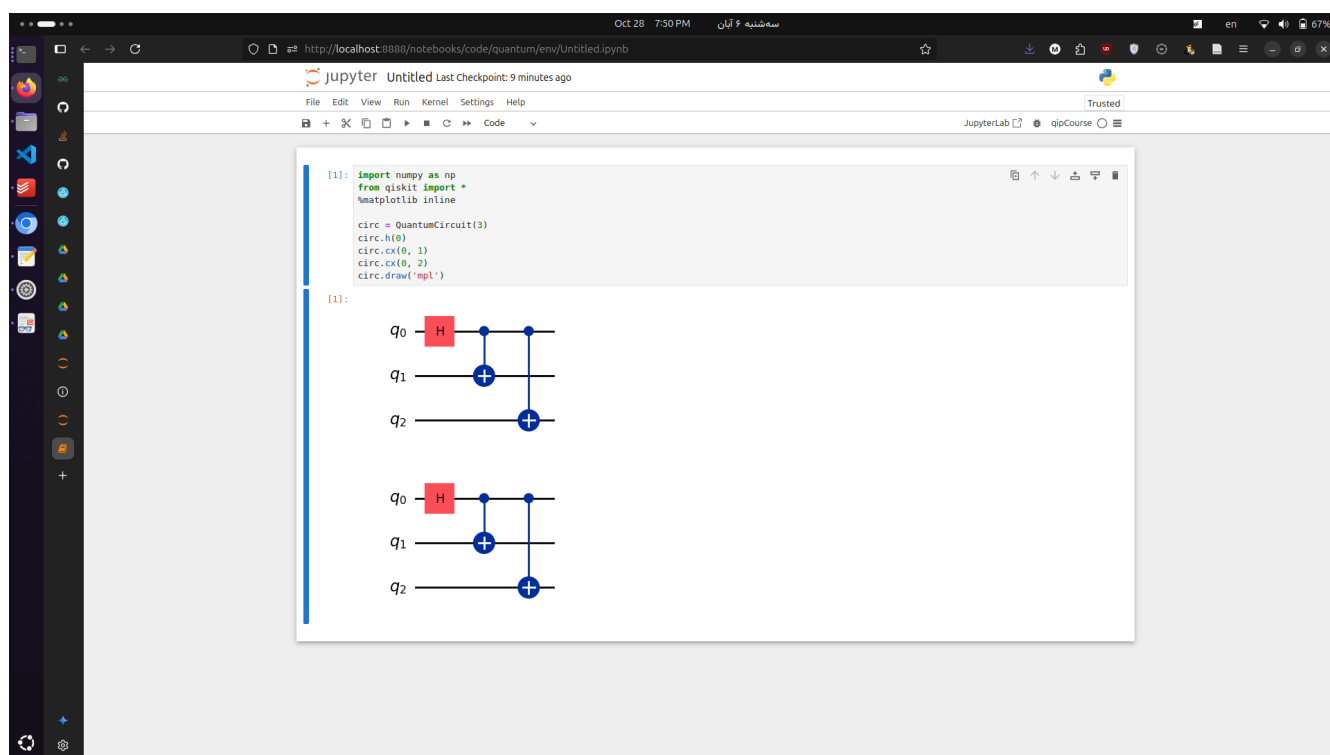
۱. یک گیت هادامارد ( $h$ ) روی کیوبیت ۰ اعمال می‌کند تا آن را به حالت سوپروپوزیشن  $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$  ببرد.

۲. یک گیت  $CNOT$  (یا  $cx$ ) از کیوبیت کنترل ۰ به کیوبیت هدف ۱ اعمال می‌کند.

۳. یک گیت  $CNOT$  دیگر از کیوبیت کنترل ۰ به کیوبیت هدف ۲ اعمال می‌کند.

این مجموعه از گیت‌ها، یک حالت درهم‌تنیده‌ی سه کیوبیتی معروف به حالت  $GHZ$  (GreenbergerHorneZeilinger) ایجاد می‌کند.

دستور نهایی، `circ.draw('mpl')`، از matplotlib برای ترسیم شماتیک این مدار استفاده کرده که خروجی آن در شکل ۱ قابل مشاهده است.



شکل ۱: اجرا شبیه سازی و ترسیم مدار  $GHZ$