# C:\Users\RAVEL\Desktop\RAVEL\Desktop\6to Semestre\xotros\tec logo.jpgJuliana Peña A01165536 Esteban Pérez A01163982 Ravel Domínguez A01165537 Renzo Cangahuala A01164025

**TC1013.1  
Métodos Numéricos y Álgebra Lineal  
Reporte del Proyecto Final**

# Computación cuántica

## La computación cuántica, a diferencia de la computación clásica se basa en el uso de la mecánica cuántica por medio de qubits. Esto es con el propósito de encontrar nuevos algoritmos que usen principios de la mecánica cuántica, con el objetivo de dar diferentes caminos de solución a problemas difíciles de resolver mediante la computación clásica.

## Qubit

Un **qubit**  *(quantum bit)* es el equivalente cuántico del bit clásico. Un qubit puede tener un valor de o , los estados bases, o una superposición cuántica de ellos. Esto quiere decir, un qubit se puede representar de la siguiente manera:

Eso es, un qubit es un vector unitario donde y son números complejos y las magnitudes de probabilidad de cada posible estado.

## Compuertas de 1 qubit

Los qubits se pueden modificar con compuertas cuánticas, equivalentes a las compuertas lógicas del cómputo clásico. Las compuertas cuánticas son transformaciones que se aplican sobre la representación vectorial del qubit. Las compuertas de 1 qubit son matrices unitarias de 2x2. es unitaria si:

Donde es la transpuesta conjugada de . La compuerta debe ser unitaria porque este es el único tipo de matriz que conserva la propiedad de .

Algunos ejemplos de compuertas cuánticas son:

### Compuerta identidad (no hacer nada)

### Compuerta X (not cuántico)

### Compuerta Z

### Compuerta Hadamard

# Dos qubits

En un sistema de dos qubits, hay cuatro estados bases. Un sistema de dos qubits se puede representar así:

O también mediante su representación vectorial:

Tomando en cuenta las siguientes restricciones:

### Not controlado (c-not), compuerta de dos bits

Las compuertas de 2 qubits son matrices de 4x4. La más importante es el Not controlado, o c-not:

El primer qubit es el control y el segundo el objetivo. Al objetivo se le aplica un not si el control es 1.

**El cómputo cuántico es universal con compuertas de un qubit y c-not**

# Protocolo de superdense coding

Este protocolo es una aplicación del cómputo cuántico; permite transmitir dos bits clásicos en un qubit y funciona de la siguiente manera.

Eve tiene dos qubits en el estado . Eve aplica una compuerta Hadamard en el primer qubit y una compuerta c-not en los dos qubits para producir el entrelazamiento (estado Bell)

Eve le da uno de sus qubits a Alice y otro a Bob.

Alice tiene dos bits clásicos que quiere mandar a Bob con sólo un qubit. Alice aplica diferentes compuertas para almacenar sus bits clásicos:

|  |  |  |
| --- | --- | --- |
| Bits clásicos | Operación cuántica | Estado resultante |
| 00 |  |  |
| 01 |  |  |
| 10 |  |  |
| 11 |  |  |

Alice envía su qubit modificado a Bob. Bob reversa las operaciones de Eve (primero c-not y después Hadamard) para producir:

|  |  |
| --- | --- |
| Estado inicial | Estado final |
|  |  |
|  |  |
|  |  |
|  |  |

El cual es el estado original de los bits clásicos de Alice.

Esto se puede hacer gracias a que las operaciones de Alice afectan también al qubit de Bob por el entrelazamiento cuántico que existe entre ellos.

Para este proyecto, programamos el protocolo de superdense coding en Python, junto con representaciones de sistemas de uno y dos qubits.

# Otras aplicaciones

Otras aplicaciones del cómputo cuántico son las siguientes:

**Encriptación:** codificar información prácticamente imposible de romper por personas ajenas.

**Teletransportación:** comunicar el estado físico de un objeto a otro en distinto lugar

**La factorización de grandes números:** reducción del tiempo de proceso de la operación de factorización de números grandes.

**La búsqueda en bases de datos:**Reducción del tiempo de búsqueda de elementos de una base de datos. Actualmente el promedio de los algoritmos tiene una complejidad de mientras que la computación cuántica nos permite bajar esa complejidad hasta obtener una complejidad de .

# Código

El código incluido en este proyecto consta de dos partes.

El archivo qubit.py es un simulador de cómputo cuántico básico para sistemas de uno y dos qubits, e incluye compuertas básicas como X, Z, Hadamard y CNOT, además de métodos para representar y medir el qubit.

El archivo superdense.py es una implementación del protocolo de superdense coding usando el simulador del archivo qubit.py. superdense.py es ejecutable y corre el protocolo al cargarse.

El código depende de Python 2.7 y no funciona con Python 3.

# Fuentes

* Mnielsencourses. (2010, Diciembre 22). *Quantum Computing for the Determined*. Recuperado el 3 de Mayo de 2010 de: <http://www.youtube.com/user/mnielsencourses>
* *Superdense coding*. (2010, Agosto 27). Recuperado el 3 de Mayo de 2010 de: <http://en.wikipedia.org/wiki/Superdense_coding>
* *Qubit*. (2011, Abril 20). Recuperado el 3 de Mayo de 2010 de: <http://es.wikipedia.org/wiki/Qubit>
* *Computación cuántica*. (2011, Abril 22). Recuperado el 3 de Mayo de 2010 de: <http://es.wikipedia.org/wiki/Computación_cuántica>
* Chávez, J. (2004, Junio 30). *Computación cuántica*. Recuperado el 3 de Mayo de 2010 de: http://www.enterate.unam.mx/Articulos/2004/junio/cuantica.htm