Operator Module API

Unitary operators can be constructed via the operator module.

In FQAM, for now, all operators are specified by their outer product representation in the following manner:

$$\sqrt{\mathbf{NOT}} = \begin{cases} U_{00} := \frac{1}{\sqrt{2}} \left|0\right\rangle \left\langle 0\right| \\ U_{01} := \frac{i}{\sqrt{2}} \left|0\right\rangle \left\langle 1\right| \\ U_{10} := \frac{i}{\sqrt{2}} \left|1\right\rangle \left\langle 0\right| \\ U_{11} := \frac{1}{\sqrt{2}} \left|1\right\rangle \left\langle 1\right| \end{cases}$$

Which corresponds to the following in the codebase:

```
FQAM_Operator_add_term(FQAM_op *operator, Complex alpha, FQAM_op *op_term);
```

Args:

- FQAM_op *operator: A pointer to an operator object to add to.
- Complex alpha: Complex scalar coefficient.
- FQAM_op op_term: Operator object representing the term.

Example:

```
FQAM_Operator_add_term(FQAM_op operator, 1/sqrt(2), |0><0|);
FQAM_Operator_add_term(FQAM_op operator, i/sqrt(2), |1><0|);
FQAM_Operator_add_term(FQAM_op operator, i/sqrt(2), |0><1|);
FQAM_Operator_add_term(FQAM_op operator, 1/sqrt(2), |1><1|);
```

where each $\left|0\right\rangle$ and $\left|1\right\rangle$ are part of some orthonormal basis.

Aside:

In general, I am not too sure yet about the theory of operator decomposition, but in the future, I would like to be able to add functionality such that any unitary operator A can be specified in terms of a decomposition of other operators. The outer product case above can be thought of as a special case of this.

More formally, the user should be able to define an operator A like so:

$$A = \sum_{i=1}^{n} \alpha_i P_i, \text{ where } P_i \text{ is an operator}$$

Generating Orthonormal Basis

An orthonormal basis in Hilbert space C^2 is parameterized by:

$$\left\{ \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix}, \begin{bmatrix} \sin(\theta) \\ -\cos(\theta) \end{bmatrix} \right\}$$

Hence in the code base one can generate any

```
FQAM_Operator_outer_product (FQAM_Op *op, int theta);
```

Args:

- FQAM_Op *op: Operator store result into
- int theta: Angle in [0, 2pi].