

Формула для расчёта

Общее значение для услуг в одной задаче – состояние кубита (исходное – нулевое)

Преобразования зависят от применённой карточки (формулы описаны в изображении гейта)

Adminer

The screenshot shows the Adminer interface for PostgreSQL. The top bar indicates the connection is to 127.0.0.1:8081, the database is 'public' in 'postgres', and the schema is 'public'. The main title is 'Схема: public'. Below the title, there are tabs for 'Изменить схему', 'Схема базы данных', 'Хранимые процедуры и функции', '«Последовательности»', and 'Типы пользователей'. A search bar for tables is present. The main area displays a table of tables with columns: Таблица (Table), Тип таблиц (Table type), Режим сопоставления (Matching mode), Объём данных? (Data volume?), Объём индексов? (Index volume?), Свободное место (Free space), Автоматическое приращение (Automatic extension), Строк? (Rows?), and Комментарий? (Comment?). The table lists four tables: 'degrees_to_gates' (type table), 'gates' (type table), 'quantum_tasks' (type table), and 'users' (type table). Below the table, there are buttons for 'Вакум' (Vacuum), 'Оптимизировать' (Optimize), 'Очистить' (Clear), and 'Удалить' (Delete). Further down, there are buttons for 'Создать таблицу' (Create table) and 'Создать представление' (Create view). The bottom section shows sequences with the heading '«Последовательности»' and lists 'gates_id_gate_seq' and 'quantum_tasks_id_task_seq'.

Добавление услуги

The screenshot shows the Adminer interface for PostgreSQL. The top bar indicates the connection is to 127.0.0.1:8081, the database is 'public' in 'postgres', and the schema is 'public'. The main title is 'SQL-запрос'. The left sidebar shows the schema 'public' and lists 'SQL-запрос', 'Импорт', 'Экспорт', and 'Создать таблицу'. Below the sidebar, there is a list of available tables: 'degrees_to_gates', 'gates', 'quantum_tasks', and 'users'. The main area contains an SQL insert statement:

```
INSERT INTO public.gates (
    id_gate,
    title,
    description,
    status,
    full_info,
    the_axis
) VALUES
(
    7,
    'Random Gate',
    'Вымышленный гейт для проверки работоспособности программы.',
    true,
    'Ничего не делает с состоянием кубита. Оставляет его без изменений.',
    'non'
)
```

A green message at the bottom states: 'Запрос завершён, изменена 1 запись. (0.005 s) Редактировать' (Query completed, 1 record changed. (0.005 s) Edit).

Просмотр результатов добавления

The screenshot shows the Adminer interface for PostgreSQL. The top bar indicates the connection is to 127.0.0.1:8081, the database is 'public' in 'postgres', and the schema is 'public'. The main title is 'SQL-запрос'. The left sidebar shows the schema 'public' and lists 'SQL-запрос', 'Импорт', 'Экспорт', and 'Создать таблицу'. Below the sidebar, there is a list of available tables: 'degrees_to_gates', 'gates', 'quantum_tasks', and 'users'. The main area shows the result of a SELECT query on the 'gates' table:

id_gate	title	description	status	image
1	Identity Gate	Не изменяет состояния кубита.	1	http://127.0.0.1:9000/lbm-pictures/img/I-gate.png
2	Pauli-X Gate (NOT gate)	Инвертирует состояние кубита.	1	http://127.0.0.1:9000/lbm-pictures/img/X-gate.png
3	X-axis Rotation Gate	Вращает кубит вокруг оси X на угол тэта.	1	http://127.0.0.1:9000/lbm-pictures/img/X-rot-gate.png
4	Y-axis Rotation Gate	Вращает кубит вокруг оси Y на угол тэта.	1	http://127.0.0.1:9000/lbm-pictures/img/Y-rot-gate.png
5	Z-axis Rotation Gate	Вращает кубит вокруг оси Z на угол тэта.	1	http://127.0.0.1:9000/lbm-pictures/img/Z-rot-gate.png
6	H (Hadamard) Gate	Создает равномерную суперпозицию из базисного состояния.	1	http://127.0.0.1:9000/lbm-pictures/img/H-gate.png
7	Random Gate	Вымышленный гейт для проверки работоспособности программы.	1	NULL

Below the table, it says '7 строк (0.001 s) Редактировать, Explain, Экспорт' (7 rows (0.001 s) Edit, Explain, Export).

3-и страницы приложения

localhost:8080 IBM quantum

Выберите однокубитную операцию

Введите название гейта Найти 3

Identity Gate
Не изменяет состояния кубита.

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Pauli-X Gate (NOT gate)
Инвертирует состояние кубита.

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

X-axis Rotation Gate
Вращает кубит вокруг оси X на угол θ .

$$\begin{bmatrix} \cos \frac{\theta}{2} & -i \sin \frac{\theta}{2} \\ -i \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{bmatrix}$$

Y-axis Rotation Gate
Вращает кубит вокруг оси Y на угол θ .

$$\begin{bmatrix} \cos \frac{\theta}{2} & \sin \frac{\theta}{2} \\ \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{bmatrix}$$

localhost:8080 IBM quantum gate

Подробнее об однокубитном гейте

Identity Gate
Ничего не делает с состоянием кубита. Оставляет его без изменений.

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

localhost:8080 IBM quantum list

Описание
Добавьте описание задачи, чтобы не перепутать её с другими.
Компания АББГД. Задача номер 1. Оправдание статистических гипотез.

Укажите детали выражения
Для этого расположите карточки в необходимом порядке и введите углы для операций вращения.

Identity Gate $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

Pauli-X Gate (NOT gate) $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$

X-axis Rotation Gate $\begin{bmatrix} \cos \frac{\theta}{2} & -i \sin \frac{\theta}{2} \\ -i \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{bmatrix}$

Y-axis Rotation Gate $\begin{bmatrix} \cos \frac{\theta}{2} & \sin \frac{\theta}{2} \\ \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{bmatrix}$ $\theta = 30$

Результат
0.2588|0⟩ + 0.9659|1⟩

Поиск

The screenshot shows the IBM Quantum search interface at localhost:8080. A search bar at the top contains the text "Identity Gate". Below it, a search result for "Identity Gate" is displayed, showing its matrix form $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. To the right is a photograph of a quantum computing hardware component labeled "IBM Quantum".

Удаление заявки

The screenshot shows the IBM Quantum search interface at localhost:8080. A search bar at the top contains the text "введите название гейта". Below it, several gate operations are listed for deletion:

- Identity Gate**: Не изменяет состояния кубита. Matrix: $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
- Pauli-X Gate (NOT gate)**: Инвертирует состояние кубита. Matrix: $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
- X-axis Rotation Gate**: Вращает кубит вокруг оси X на угол тэта. Matrix: $\begin{bmatrix} \cos \frac{\theta}{2} & -i\sin \frac{\theta}{2} \\ -i\sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{bmatrix}$
- Y-axis Rotation Gate**: Вращает кубит вокруг оси Y на угол тэта. Matrix: $\begin{bmatrix} \cos \frac{\theta}{2} & -\sin \frac{\theta}{2} \\ \sin \frac{\theta}{2} & \cos \frac{\theta}{2} \end{bmatrix}$

Проверка удаления

The screenshot shows a browser window at localhost:8080/quantum_task/1. The response body is a JSON object: {"description": "task not found or has been deleted", "status": "error"}

The screenshot shows the Adminer database interface at 127.0.0.1:8081. The connection is to PostgreSQL » postgres » quantum » public » Выбрать: quantum_tasks. The table structure is shown with columns: id_task, task_status, creation_date, id_user, conclusion_date, id_moderator. A single row is selected for editing, showing id_task = 1, task_status = удален, creation_date = 2025-09-29 07:51:48.847007+00, id_user = 1, conclusion_date = NULL, id_moderator = NULL.

Добавление услуг в заявку

localhost:8080 IBM quantum list

Описание

Добавьте описание задачи, чтобы не перепутать её с другими.

Укажите детали выражения

Для этого расположите карточки в необходимом порядке и введите углы для операций вращения.

Pauli-X Gate [0 1] [1 0]	Pauli-X Gate (NOT gate)
Identity Gate [1 0] [0 1]	Identity Gate
Hadamard Gate [1/2 1/2] [1/2 -1/2]	H (Hadamard) Gate
Z-axis Rotation Gate [cosθ -sinθ] [sinθ cosθ]	Z-axis Rotation Gate $\theta = 30$
Y-axis Rotation Gate [cosθ -sinθ] [sinθ cosθ]	Y-axis Rotation Gate $\theta = 40$

Удалить задачу

Результатат

Проверка

127.0.0.1:8081 PostgreSQL > postgres > quantum > public > Выбрать: quantum_tasks

Выбрать: quantum_tasks

	Выбрать	Показать структуру	Изменить таблицу	Новая запись				
	Выбрать	Поиск	Сортировать	Лимит 50 100 Длина текста				
				Действие Выбрать				
	SELECT * FROM "quantum_tasks" LIMIT 50 (0.001 s) Редактировать							
	<input type="checkbox"/> Изменить	id_task	task_status	creation_date	id_user	conclusion_date	id_moderator	task_description
	<input type="checkbox"/> редактировать	1	удален	2025-09-29 07:51:48.847007+00	1	NULL	NULL	Компания АБВГД. Задача номер 1. Оправдание
	<input type="checkbox"/> редактировать	2	удален	2000-01-01 00:00:00+00	1	0001-01-01 00:00:00+00	NULL	
	<input type="checkbox"/> редактировать	3	удален	2000-01-01 00:00:00+00	1	0001-01-01 00:00:00+00	NULL	
	<input type="checkbox"/> редактировать	4	удален	2000-01-01 00:00:00+00	1	0001-01-01 00:00:00+00	NULL	
	<input type="checkbox"/> редактировать	5	удален	2000-01-01 00:00:00+00	1	0001-01-01 00:00:00+00	NULL	
	<input type="checkbox"/> редактировать	6	черновик	2000-01-01 00:00:00+00	1	0001-01-01 00:00:00+00	NULL	

Весь результат 6 строк Сохранить Редактировать Клонировать Стереть Экспорт (6) Импорт

Изменение м-м таблицы из БД

127.0.0.1:8081 PostgreSQL > postgres > quantum > public > SQL-запрос

SQL-запрос - postgres - Adminer

Язык: Русский

DB: quantum Схема: public

SQL-запрос Импорт Экспорт Создать таблицу

выбрать degrees_to_gates
выбрать gates
выбрать quantum_tasks
выбрать users

UPDATE degrees_to_gates SET degrees = 10 WHERE id_gate = 5 AND id_task = 6

Запрос завершён, изменена 1 запись. (0.006 s) Редактировать

UPDATE degrees_to_gates SET degrees = 10 WHERE id_gate = 5 AND id_task = 6

Изменение таблицы заявки из БД

The screenshot shows the Adminer interface. In the top right, it says "Закрыть" (Close) and "SQL-запрос - postgres - Adminer". Below that, "Язык: Русский" (Language: Russian) and "PostgreSQL » postgres » quantum » public » SQL-запрос" are displayed. The main area is titled "SQL-запрос" (SQL Query). A green status bar at the bottom says "Запрос завершён, изменена 1 запись. (0.006 s) Редактировать" (Query completed, 1 record changed. (0.006 s) Edit). The query itself is: `UPDATE quantum_tasks SET task_description = 'Случайный набор гейтов для проверки лабы' WHERE id_task = 6`. On the left sidebar, under "SQL-запрос", there are buttons for Импорт (Import), Экспорт (Export), and Создать таблицу (Create Table). Below these are dropdown menus for DB: quantum and Схема: public. A list of tables is shown: выбрать degrees_to_gates, выбрать gates, выбрать quantum_tasks, and выбрать users.

Результат изменений

The screenshot shows the "IBM quantum list" application running on localhost:8080. The title bar says "localhost:8080" and "IBM quantum list". The main area has a heading "Описание" (Description) with a text input field containing "Случайный набор гейтов для проверки лабы". To the right is a "Удалить задачу" (Delete task) button. Below this is a section titled "Укажите детали выражения" (Specify expression details) with instructions: "Для этого расположите карточки в необходимом порядке и введите углы для операций вращения." (For this, place the cards in the required order and enter angles for rotation operations.). It shows two card slots: "Identity Gate" (with matrix $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$) and "Z-axis Rotation Gate" (with matrix $\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$ and angle $\theta = 10$). To the right is a cartoon illustration of a person working on a computer. The result section shows the output "1|0) + 0|1)".

Демонстрация кода

Модели

The screenshot shows two code editors. The left editor displays Go code for the "DegreesToGates" model:front_start > internal > app > ds > DegreesToGates.go ...
1 package ds
2
3 type DegreesToGates struct {
4 ID_gate uint `gorm:"primaryKey;column:id_gate;not null"
5 ID_task uint `gorm:"primaryKey;column:id_task;not null"
6 Degrees *float32
7 // Связи, необходимые для работы сервиса
8 Gate Gate `gorm:"foreignKey:ID_gate;references:ID_gate"
9 Task QuantumTask `gorm:"foreignKey:ID_task;references:ID_task"
10 }
11 The right editor displays PostgreSQL code for the "Gate" model:front_start > internal > app > ds > Gate.go > Degrees
1 package ds
2
3 type Gate struct {
4 ID_gate uint `gorm:"column:id_gate;primaryKey"
5 Title string `gorm:"column:title;size:255;not null;default:gate-no-name;unique"
6 Description string `gorm:"column:description;not null"
7 Status bool `gorm:"column:status; default:true"
8 Image *string `gorm:"column:image"
9
10 // subject area
11 FullInfo string `gorm:"column:full_info;not null"
12 TheAxis string `gorm:"column:the_axis"
13 /*
14 Удалил согласно правкам к ЛР 1
15 IsEditable bool `gorm:"default:false"
16 */
17
18 // Несущая связь
19 Degrees []DegreesToGates `gorm:"foreignKey:ID_gate;references:ID_gate"
20 }

Т. М-м

Т. УСЛУГИ

```

front_start > internal > app > ds > ❷❸ QuantumTask.go > ❹❺ QuantumTask
1 package ds
2
3 import "time"
4
5 type QuantumTask struct {
6     ID_task      uint      `gorm:"column:id_task;primaryKey;autoIncrement"`
7     TaskStatus   string    `gorm:"column:task_status;size:255;not null;default:черновик"`
8     CreationDate time.Time `gorm:"column:creation_date;not null"`
9     ID_user      uint      `gorm:"column:id_user;not null"`
10    ConclusionDate time.Time `gorm:"column:conclusion_date"`
11    ID_moderator *uint     `gorm:"column:id_moderator"`
12
13    TaskDescription string    `gorm:"column:task_description"`
14    Res_koeff_0 float32   `gorm:"column:res_koeff_0;default:1"`
15    Res_koeff_1 float32   `gorm:"column:res_koeff_1;default:0"`
16
17    // Несущая связь
18    GatesDegrees []DegreeesToGates `gorm:"foreignKey:ID_task;references:ID_task"`
19    Moderator   Users      `gorm:"foreignKey:ID_moderator;references:ID_user"`
20
21    // Связь для программы
22    User   Users      `gorm:"foreignKey:ID_user;references:ID_user"`
23 }

```

Т. ЗАКАЗЫ

Составной уникальный ключ

```

front_start > internal > app > ds > ❷❸ DegreesToGates.go > ...
1 package ds
2
3 type DegreeesToGates struct {
4     ID_gate uint `gorm:"primaryKey;column:id_gate;not null"`
5     ID_task uint `gorm:"primaryKey;column:id_task;not null"`
6     Degrees *float32
7
8     // Связи, необходимые для работы сервиса
9     Gate Gate      `gorm:"foreignKey:ID_gate;references:ID_gate"`
10    Task QuantumTask `gorm:"foreignKey:ID_task;references:ID_task"`
11 }

```

4-ре контроллера через ORM

```

func (r *Repository) GetGates() ([]ds.Gate, error) {
    var gates []ds.Gate

    err := r.db.Find(&gates).Error
    if err != nil {
        return nil, err
    }

    if len(gates) == 0 {
        return nil, fmt.Errorf("gates not found")
    }
    return gates, nil
}

func (r *Repository) GetGatesByName(title string) ([]ds.Gate, error) {
    var gates []ds.Gate
    err := r.db.Where("title ILIKE ?", "%" + title + "%").Find(&gates).Error
    if err != nil {
        return nil, err
    }
    return gates, nil
}

// AddGateToTask добавляет гейт в задачу (создает запись в таблице т-м).
func (r *Repository) AddGateToTask(taskID, gateID uint) error {
    // Проверяем, нет ли уже такого фактора в заявке, чтобы избежать дублей
    var count int64
    r.db.Model(&ds.DegreeesToGates{}).Where("id_task = ? AND id_gate = ?", taskID, gateID).Count(&count)
    if count > 0 {
        return errors.New("gate already in task")
    }

    link := ds.DegreeesToGates{
        ID_task: taskID,
        ID_gate: gateID,
    }
    return r.db.Create(&link).Error
}

func (r *Repository) GetGateByID(id int) (*ds.Gate, error) {
    var gate ds.Gate
    err := r.db.First(&gate, id).Error
    if err != nil {
        return nil, err
    }
    return &gate, nil
}

// GetTaskWithGates получает задачу со всеми связанными гейтами и их углами поворота.
func (r *Repository) GetTaskWithGates(taskID uint) (*ds.QuantumTask, error) {
    var task ds.QuantumTask

    // Используем Preload для загрузки связанных данных через связующую таблицу
    err := r.db.
        Preload("GatesDegrees.Gate").
        First(&task, taskID).
        Error

    if err != nil {
        if errors.Is(err, gorm.ErrRecordNotFound) {
            return nil, errors.New("task not found")
        }
        return nil, fmt.Errorf("error fetching task: %v", err)
    }

    // Проверяем, что задача не удалена (адаптируйте под ваши статусы)
    if task.TaskStatus == ds.StatusDeleted {
        return nil, errors.New("task not found or has been deleted")
    }

    return &task, nil
}

```

Т. ПОЛЬЗОВАТЕЛИ

```

front_start > internal > app > ds > ❷❸ Users.go > ...
1 package ds
2
3 type Users struct {
4     ID_user uint      `gorm:"column:id_user;primaryKey"`
5     Login   string    `gorm:"column:login;not null;size:255;unique"`
6     Password string    `gorm:"column:password;size:255;not null"`
7     IsAdmin  bool     `gorm:"column:is_admin;default:false"`
8
9     // Связь несущая
10    Tasks []QuantumTask `gorm:"foreignKey:ID_user;references:ID_user" json:"-"`
11 }

```

Удаление заявки через SQL

```
// LogicallyDeleteTask выполняет логическое удаление заявки через чистый SQL UPDATE.
func (r *Repository) LogicallyDeleteTask(taskID uint) error {
    // Используем Exec для выполнения "сырого" SQL-запроса
    result := r.db.Exec("UPDATE quantum_tasks SET task_status = ? WHERE id_task = ?", ds.StatusDeleted, taskID)
    return result.Error
}
```