

Задания отборочного тура Открытой Олимпиады СПбГУ среди студентов и молодых специалистов «Petropolitan Science (Re)Search» в 2024/25 учебном году по предмету «Вычислительные технологии» для обучающихся и выпускников бакалавриата

Задание 1. Необходимо создать программу, способную играть в классические шашки с пользователем. Программа должна корректно обрабатывать все правила игры, определять победителя, а также использовать параллельный алгоритм для повышения скорости выполнения.

Условия:

- Программа должна поддерживать классические правила игры в шашки: обычные и дамочные ходы, рубку шашек противника, принудительную рубку.
- Программа должна быть написана на языке C++ (стандарт не ниже 17). Разрешено использовать только стандартную библиотеку.
- Параллельный алгоритм может быть реализован с помощью многопоточности (например, `std::thread` или `openmp` в C++).

Запуск и управление:

Программа должна запускаться через консоль.

При запуске пользователю нужно предложить выбрать, за кого он хочет играть: за белых или за черных.

- Если пользователь выбрал “белых”, программа сразу же выводит игровое поле с расставленными фигурами (формат см. ниже) и ожидает ввода хода от пользователя.
- Если пользователь выбрал “черных”, программа сначала делает ход за белых, затем выводит игровое поле.

После каждого хода пользователя программа должна делать корректный ход за противоположную сторону и выводить игровое поле. Разрешается выводить название столбцов и строк в развернутом порядке. Например, HGFEDCBA и 87654321 соответственно.

.		A	B	C	D	E	F	G	H
1		.	W	.	W	.	W	.	W
2		W	.	W	.	W	.	W	.
3		.	W	.	W	.	W	.	W
4	
5	
6		B	.	B	.	B	.	B	.
7		.	B	.	B	.	B	.	B
8		B	.	B	.	B	.	B	.

Пользователь делает ход в формате A6 B5, где первая координата — начальная позиция шашки, а вторая — конечная позиция.

В качестве ответа необходимо предоставить текст с описанием алгоритма и анализом вариантов решения, а также ссылку на открытый репозиторий GitHub (или аналогичный) с кодом реализованной программы (дата последнего коммита не позже даты завершения отборочного тура олимпиады) или приложите ссылку на архив с кодом. Помните о необходимости соблюдения анонимности при предоставлении материалов.

Задание 2. Найти угол поворота изображения документа, содержащего печатный текст и графику. Скорректировать изображение путем поворота, используя найденный угол. Примеры исходного и повернутого изображения прилагаются.

В качестве ответа необходимо предоставить текст с описанием предлагаемого алгоритма, а также ссылку на открытый репозиторий GitHub (или аналогичный) с кодом реализованной программы (дата последнего коммита не позже даты завершения отборочного тура олимпиады) либо ссылку на архив с кодом. Помните о необходимости соблюдения анонимности при предоставлении материалов.

Наиболее просто регистры реализуются на основе тактируемых фронтом D-триггеров. Схема параллельного регистра приведена на рис. 16.



Рисунок 1

В ней разряды двоичного числа, поступающие, например, с выходов другого регистра, соединены с информационными входами D_1, D_2, \dots, D_n . D-триггеров, тактовые входы, которых объединены между собой и представляют собой единый тактовый вход регистра. Таким образом, после подачи импульса на тактовый вход, происходит запись информации с входов D_i на выходы Q_i . В этом случае параллельный регистр, представляет собой просто набор триггеров, объединенных общим тактовым сигналом. Схема сдвигающего регистра приведена на рис. 17.

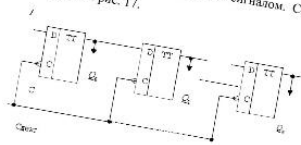


Рисунок 2

Наиболее просто регистры реализуются на основе тактируемых фронтом D-триггеров. Схема параллельного регистра приведена на рис. 16.

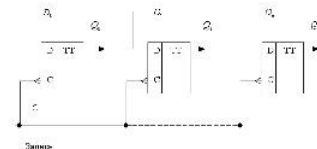


Рисунок 1

В ней разряды двоичного числа, поступающие, например, с выходов другого регистра, соединены с информационными входами D_1, D_2, \dots, D_n . D-триггеров, тактовые входы, которых объединены между собой и представляют собой единый тактовый вход регистра. Таким образом, после подачи импульса на тактовый вход, происходит запись информации с входов D_i на выходы Q_i . В этом случае параллельный регистр, представляет собой просто набор триггеров, объединенных общим тактовым сигналом. Схема сдвигающего регистра приведена на рис. 17.

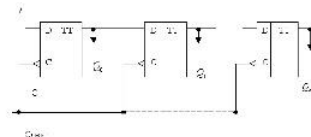


Рисунок 2

Задание 3. Для защиты данных используют разные методы или техники, начиная от простых шифров до гомоморфного шифрования. Многие считают, что для защиты персональных данных хватит и обычных шифров, так как другие методы, такие как AES-128, SHA-512 и т.п. очень сложны или требуют существенных вычислительных ресурсов.

Ваша задача – определить алгоритм, с помощью которого зашифрованы представленные ниже данные; написать программу, предпочтительно используя язык программирования Python, которая деобезличивает этот набор данных.

email	Адрес
pmjmz.bzmujtig@nzqmamv.kwu	ыу. Лпмшорхщтцкцл.6 тй.466
giyh.lsxyl@pifeguh.vct	Юдубвудашэюут жя.ч.34 юх.102
enxbepbms.tkexgx@ykbxlg.vhf	вг. Сэёкэятц.60 эф.107
eobh.хурух@zyyhys.wig	Ф. Аудзьбеюут жя.ч.31 юх.159
kizumtw36@rwpvawv.kwu	ыу. Игщъшцунъцйзл.81 тй.107
nevvix.kpiewsr@lsxqemp.gsqa	Ёяхтодг чп.з.77 оё.232
pexkxexk.biblklob@elqjxfi.zlj	Дешезйпъшзбцх йв.ъ.80 бш.422
ugyvp@wbkj.xod	эх. Нйбчийн.25 фл.43
trwxg.gbxxghp@tulabkx.ubs	Дебюубфтс ёю.ц.14 эф.61
tdquvwjw.wjxvrn@hjxx.lxv	Ъщ. Усьфчкьуст шнщ.м.73 ук.65

В качестве ответа предоставить объяснение способа определения загаданного алгоритма обезличивания данных и назвать сам алгоритм, представить краткое описание реализованного решения, блок-схему алгоритма решения, результат работы программы: деобезличенный датасет с добавлением столбца, в котором указан ключ шифрования; предоставить ссылку на открытый репозиторий GitHub (или аналогичный) с кодом реализованной программы (дата последнего коммита не позже даты завершения отборочного тура олимпиады) либо ссылку на архив с кодом. Помните о необходимости соблюдения анонимности при предоставлении материалов.

Критерии оценки решений задач

Задание 1. Общий балл за решение задачи выставляется как сумма баллов за соответствие решения каждому из критериев.

I. Корректность правил игры (до 7 баллов):

- Программа должна обрабатывать все основные правила: обычные ходы, рубку, превращение в дамки, завершение игры и т.д.
- Ошибки в правилах, например, некорректное выполнение рубки или игнорирование правил, будут снижать балл.

II. Оптимизация работы программы (до 4 баллов):

- Программа должна работать эффективно, без значительных задержек при обработке ходов.
- 4 балла — обработка хода занимает менее 0.5 секунды.
- 2 балла — время на ход 0.5–2 секунды.
- 0 баллов — ход занимает более 2 секунд.

III. Параллельная реализация алгоритма (до 4 баллов):

- Реализация параллельного алгоритма для ускорения расчета ходов.
- 4 балла — программа корректно использует параллельные вычисления, распределяя поиск по нескольким потокам.
- 2 балла — параллельные вычисления реализованы, но неэффективны или вызывают конфликты данных.
- 0 баллов — параллельность отсутствует.

IV. Тестирование и обработка ошибок (до 2 баллов):

- Наличие обработки некорректных данных, таких как недопустимые ходы.
- 2 балла — программа стабильно работает и корректно обрабатывает все виды ходов.
- 1 балл — программа имеет незначительные ошибки, но они не приводят к сбоям.
- 0 баллов — ошибки критичны и мешают нормальной работе программы.

Задание 2. Общий балл за решение задачи выставляется как сумма баллов за соответствие решения каждому из критериев.

I. Выбраны подходящие алгоритмы для определения угла и проведения поворота изображения, приведено их описание (до 5 баллов)

II. Представлено обоснование предложенного решения с точки зрения работы алгоритмов, занимаемой памяти и используемых ресурсов (до 4 баллов)

III. Представлен программный код, реализующий алгоритмы (до 4 баллов)

IV. Проведена экспериментальная оценка точности работы алгоритма с использованием программного решения. Оценка проводится на нескольких изображениях (не менее трех) различных документов, содержащих фото в разных местах документов. (до 4 баллов)

Задание 3. Общий балл за решение задачи выставляется как сумма баллов за соответствие решения каждому из критериев.

- I. Правильно определен использованный алгоритм обезличивания (до 2 баллов)
- II. Представлена корректная блок-схема алгоритма деобезличивания (до 4 баллов)
- III. Представлен программный код, реализующий предложенную задачу (до 5 баллов)
- IV. Программа выводит корректный результат: файл с деобезличенным набором данных (до 3 баллов)
- V. В итоговом файле выведен столбец, где в каждой строке указан верный ключ шифрования (до 2 баллов)

Максимальное количество баллов, которое можно получить за выполнение заданий:

Задание 1 – 17 баллов.

Задание 2 – 17 баллов.

Задание 3 – 16 баллов.

Максимальный балл за всю работу – 50 баллов.

Наиболее просто регистры реализуются на основе тактируемых фронтом D-триггеров. Схема параллельного регистра приведена на рис. 16.

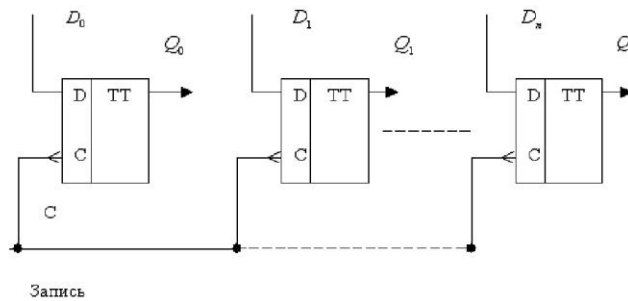


Рисунок 1

В ней разряды двоичного числа, поступающие, например, с выходов другого регистра, соединены с информационными входами D_0, D_1, \dots, D_n D-триггеров, тактовые входы, которых объединены между собой и представляют собой единый тактовый вход регистра. Таким образом, после подачи импульса на тактовый вход, происходит запись информации с входов D_i на выходы Q_i . В этом случае параллельный регистр, представляет собой просто набор триггеров, объединенных общим тактовым сигналом. Схема сдвигающего регистра приведена на рис. 17.

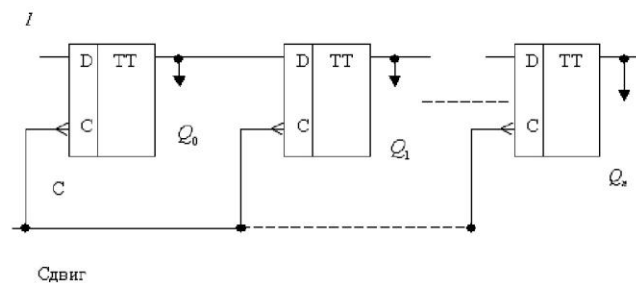


Рисунок 2

Наиболее просто регистры реализуются на основе тактируемых фронтом D-триггеров. Схема параллельного регистра приведена на рис. 16.

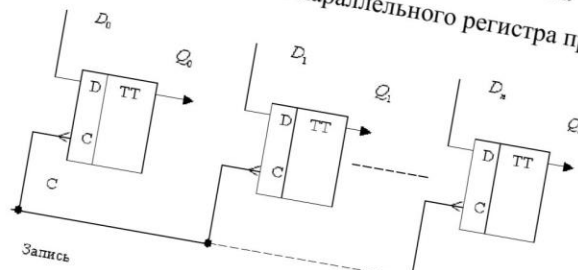


Рисунок 1

В ней разряды двоичного числа, поступающие, например, с выходов другого регистра, соединены с информационными входами D_0, D_1, \dots, D_n D-триггеров, тактовые входы, которых объединены между собой и представляют собой единый тактовый вход регистра. Таким образом, после подачи импульса на тактовый вход, происходит запись информации с входов D_i на выходы Q_i . В этом случае параллельный регистр, представляет собой просто набор триггеров, объединенных общим тактовым сигналом. Схема сдвигающего регистра приведена на рис. 17.

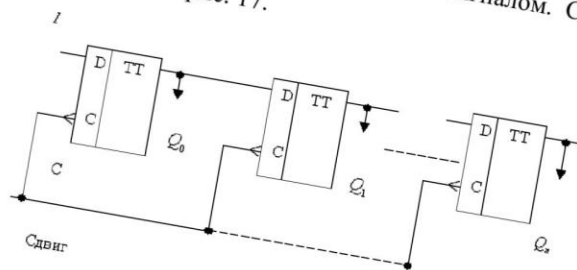


Рисунок 2