

Обробка зображень і мультимедіа

Олег Гутік



Лекція 24: Стиснення зображень, XVIII. JPEG. Квантування

Після обчислення всіх коефіцієнтів DCT їх потрібно проквантувати. На цьому етапі відбувається відкидання частини інформації (невеликі втрати відбуваються і на попередньому етапі через кінцеву точність обчислень на комп'ютері). Кожне число з матриць коефіцієнтів DCT ділиться на спеціальне число з таблиці квантування, а результат округляється до найближчого цілого числа. Як зазначалося, необхідно мати такі таблиці для кожної колірної компоненти. Стандарт JPEG допускає використання чотирьох таблиць, і користувач може вибрати будь-яку з цих таблиць для квантування компонентів кольору. Усі 64 числа таблиці квантування є параметрами JPEG. В принципі, користувач може змінити будь-який коефіцієнт для досягнення більшого ступеня стиснення. На практиці дуже складно експериментувати з такою кількістю параметрів, а тому програмне забезпечення JPEG використовує два підходи:

Після обчислення всіх коефіцієнтів DCT їх потрібно проквантувати. На цьому етапі відбувається відкидання частини інформації (невеликі втрати відбуваються і на попередньому етапі через кінцеву точність обчислень на комп'ютері). Кожне число з матриць коефіцієнтів DCT ділиться на спеціальне число з таблиці квантування, а результат округляється до найближчого цілого числа. Як зазначалося, необхідно мати такі таблиці для кожної колірної компоненти. Стандарт JPEG допускає використання чотирьох таблиць, і користувач може вибрати будь-яку з цих таблиць для квантування компонентів кольору. Усі 64 числа таблиці квантування є параметрами JPEG. В принципі, користувач може змінити будь-який коефіцієнт для досягнення більшого стиснення. На практиці дуже складно експериментувати з такою кількістю параметрів, а тому програмне забезпечення JPEG використовує два підходи:

Після обчислення всіх коефіцієнтів DCT їх потрібно проквантувати. На цьому етапі відбувається відкидання частини інформації (невеликі втрати відбуваються і на попередньому етапі через кінцеву точність обчислень на комп'ютері). Кожне число з матриць коефіцієнтів DCT ділиться на спеціальне число з таблиці квантування, а результат округляється до найближчого цілого числа. Як зазначалося, необхідно мати такі таблиці для кожної колірної компоненти. Стандарт JPEG допускає використання чотирьох таблиць, і користувач може вибрати будь-яку з цих таблиць для квантування компонентів кольору. Усі 64 числа таблиці квантування є параметрами JPEG. В принципі, користувач може змінити будь-який коефіцієнт для досягнення більшого ступеня стиснення. На практиці дуже складно експериментувати з такою кількістю параметрів, а тому програмне забезпечення JPEG використовує два підходи:

Після обчислення всіх коефіцієнтів DCT їх потрібно проквантувати. На цьому етапі відбувається відкидання частини інформації (невеликі втрати відбуваються і на попередньому етапі через кінцеву точність обчислень на комп'ютері). Кожне число з матриць коефіцієнтів DCT ділиться на спеціальне число з таблиці квантування, а результат округляється до найближчого цілого числа. Як зазначалося, необхідно мати такі таблиці для кожної колірної компоненти. Стандарт JPEG допускає використання чотирьох таблиць, і користувач може вибрати будь-яку з цих таблиць для квантування компонентів кольору. Усі 64 числа таблиці квантування є параметрами JPEG. В принципі, користувач може змінити будь-який коефіцієнт для досягнення більшого стиснення. На практиці дуже складно експериментувати з такою кількістю параметрів, а тому програмне забезпечення JPEG використовує два підходи:

Після обчислення всіх коефіцієнтів DCT їх потрібно проквантувати. На цьому етапі відбувається відкидання частини інформації (невеликі втрати відбуваються і на попередньому етапі через кінцеву точність обчислень на комп'ютері). Кожне число з матриць коефіцієнтів DCT ділиться на спеціальне число з таблиці квантування, а результат округляється до найближчого цілого числа. Як зазначалося, необхідно мати такі таблиці для кожної колірної компоненти. Стандарт JPEG допускає використання чотирьох таблиць, і користувач може вибрати будь-яку з цих таблиць для квантування компонентів кольору. Усі 64 числа таблиці квантування є параметрами JPEG. В принципі, користувач може змінити будь-який коефіцієнт для досягнення більшого стиснення. На практиці дуже складно експериментувати з такою кількістю параметрів, а тому програмне забезпечення JPEG використовує два підходи:

Після обчислення всіх коефіцієнтів DCT їх потрібно проквантувати. На цьому етапі відбувається відкидання частини інформації (невеликі втрати відбуваються і на попередньому етапі через кінцеву точність обчислень на комп'ютері). Кожне число з матриць коефіцієнтів DCT ділиться на спеціальне число з таблиці квантування, а результат округляється до найближчого цілого числа. Як зазначалося, необхідно мати такі таблиці для кожної колірної компоненти. Стандарт JPEG допускає використання чотирьох таблиць, і користувач може вибрати будь-яку з цих таблиць для квантування компонентів кольору. Усі 64 числа таблиці квантування є параметрами JPEG. В принципі, користувач може змінити будь-який коефіцієнт для досягнення більшого стиснення. На практиці дуже складно експериментувати з такою кількістю параметрів, а тому програмне забезпечення JPEG використовує два підходи:

Після обчислення всіх коефіцієнтів DCT їх потрібно проквантувати. На цьому етапі відбувається відкидання частини інформації (невеликі втрати відбуваються і на попередньому етапі через кінцеву точність обчислень на комп'ютері). Кожне число з матриць коефіцієнтів DCT ділиться на спеціальне число з таблиці квантування, а результат округляється до найближчого цілого числа. Як зазначалося, необхідно мати такі таблиці для кожної колірної компоненти. Стандарт JPEG допускає використання чотирьох таблиць, і користувач може вибрати будь-яку з цих таблиць для квантування компонентів кольору. Усі 64 числа таблиці квантування є параметрами JPEG. В принципі, користувач може змінити будь-який коефіцієнт для досягнення більшого ступеня стиснення. На практиці дуже складно експериментувати з такою кількістю параметрів, а тому програмне забезпечення JPEG використовує два підходи:

Після обчислення всіх коефіцієнтів DCT їх потрібно проквантувати. На цьому етапі відбувається відкидання частини інформації (невеликі втрати відбуваються і на попередньому етапі через кінцеву точність обчислень на комп'ютері). Кожне число з матриць коефіцієнтів DCT ділиться на спеціальне число з таблиці квантування, а результат округляється до найближчого цілого числа. Як зазначалося, необхідно мати такі таблиці для кожної колірної компоненти. Стандарт JPEG допускає використання чотирьох таблиць, і користувач може вибрати будь-яку з цих таблиць для квантування компонентів кольору. Усі 64 числа таблиці квантування є параметрами JPEG. В принципі, користувач може змінити будь-який коефіцієнт для досягнення більшого ступеня стиснення. На практиці дуже складно експериментувати з такою кількістю параметрів, а тому програмне забезпечення JPEG використовує два підходи:

Після обчислення всіх коефіцієнтів DCT їх потрібно проквантувати. На цьому етапі відбувається відкидання частини інформації (невеликі втрати відбуваються і на попередньому етапі через кінцеву точність обчислень на комп'ютері). Кожне число з матриць коефіцієнтів DCT ділиться на спеціальне число з таблиці квантування, а результат округляється до найближчого цілого числа. Як зазначалося, необхідно мати такі таблиці для кожної колірної компоненти. Стандарт JPEG допускає використання чотирьох таблиць, і користувач може вибрати будь-яку з цих таблиць для квантування компонентів кольору. Усі 64 числа таблиці квантування є параметрами JPEG. В принципі, користувач може змінити будь-який коефіцієнт для досягнення більшого ступеня стиснення. На практиці дуже складно експериментувати з такою кількістю параметрів, а тому програмне забезпечення JPEG використовує два підходи:

Після обчислення всіх коефіцієнтів DCT їх потрібно проквантувати. На цьому етапі відбувається відкидання частини інформації (невеликі втрати відбуваються і на попередньому етапі через кінцеву точність обчислень на комп'ютері). Кожне число з матриць коефіцієнтів DCT ділиться на спеціальне число з таблиці квантування, а результат округляється до найближчого цілого числа. Як зазначалося, необхідно мати такі таблиці для кожної колірної компоненти. Стандарт JPEG допускає використання чотирьох таблиць, і користувач може вибрати будь-яку з цих таблиць для квантування компонентів кольору. Усі 64 числа таблиці квантування є параметрами JPEG. В принципі, користувач може змінити будь-який коефіцієнт для досягнення більшого стиснення. На практиці дуже складно експериментувати з такою кількістю параметрів, а тому програмне забезпечення JPEG використовує два підходи:

Після обчислення всіх коефіцієнтів DCT їх потрібно проквантувати. На цьому етапі відбувається відкидання частини інформації (невеликі втрати відбуваються і на попередньому етапі через кінцеву точність обчислень на комп'ютері). Кожне число з матриць коефіцієнтів DCT ділиться на спеціальне число з таблиці квантування, а результат округляється до найближчого цілого числа. Як зазначалося, необхідно мати такі таблиці для кожної колірної компоненти. Стандарт JPEG допускає використання чотирьох таблиць, і користувач може вибрати будь-яку з цих таблиць для квантування компонентів кольору. Усі 64 числа таблиці квантування є параметрами JPEG. В принципі, користувач може змінити будь-який коефіцієнт для досягнення більшого ступеня стиснення. На практиці дуже складно експериментувати з такою кількістю параметрів, а тому програмне забезпечення JPEG використовує два підходи:

Після обчислення всіх коефіцієнтів DCT їх потрібно проквантувати. На цьому етапі відбувається відкидання частини інформації (невеликі втрати відбуваються і на попередньому етапі через кінцеву точність обчислень на комп'ютері). Кожне число з матриць коефіцієнтів DCT ділиться на спеціальне число з таблиці квантування, а результат округляється до найближчого цілого числа. Як зазначалося, необхідно мати такі таблиці для кожної колірної компоненти. Стандарт JPEG допускає використання чотирьох таблиць, і користувач може вибрати будь-яку з цих таблиць для квантування компонентів кольору. Усі 64 числа таблиці квантування є параметрами JPEG. В принципі, користувач може змінити будь-який коефіцієнт для досягнення більшого ступеня стиснення. На практиці дуже складно експериментувати з такою кількістю параметрів, а тому програмне забезпечення JPEG використовує два підходи:

Після обчислення всіх коефіцієнтів DCT їх потрібно проквантувати. На цьому етапі відбувається відкидання частини інформації (невеликі втрати відбуваються і на попередньому етапі через кінцеву точність обчислень на комп'ютері). Кожне число з матриць коефіцієнтів DCT ділиться на спеціальне число з таблиці квантування, а результат округляється до найближчого цілого числа. Як зазначалося, необхідно мати такі таблиці для кожної колірної компоненти. Стандарт JPEG допускає використання чотирьох таблиць, і користувач може вибрати будь-яку з цих таблиць для квантування компонентів кольору. Усі 64 числа таблиці квантування є параметрами JPEG. В принципі, користувач може змінити будь-який коефіцієнт для досягнення більшого ступеня стиснення. На практиці дуже складно експериментувати з такою кількістю параметрів, а тому програмне забезпечення JPEG використовує два підходи:

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Яскравість

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

Кольоровість

1. Таблиця квантування, прийнята за умовчанням. Дві такі таблиці, одна для компоненти яскравості (і градації сірого кольору), іншу — для хроматичних компонент, є результатом тривалого дослідження з безліччю експериментів, проведених комітетом JPEG. Вони є частиною стандарту JPEG та відтворені в табл. Видно, як коефіцієнти QC таблиць зростають під час руху з лівого верхнього кута в правий нижній кут. У цьому відображається скорочення коефіцієнтів DCT, відповідних високим просторовим частотам.

16	11	10	16	24	40	51	61	17	18	24	47	99	99	99	99
12	12	14	19	26	58	60	55	18	21	26	66	99	99	99	99
14	13	16	24	40	57	69	56	24	26	56	99	99	99	99	99
14	17	22	29	51	87	80	62	47	66	99	99	99	99	99	99
18	22	37	56	68	109	103	77	99	99	99	99	99	99	99	99
24	35	55	64	81	104	113	92	99	99	99	99	99	99	99	99
49	64	78	87	103	121	120	101	99	99	99	99	99	99	99	99
72	92	95	98	112	100	103	99	99	99	99	99	99	99	99	99

Яскравість

Кольоровість

1. Таблиця квантування, прийнята за умовчанням. Дві такі таблиці, одна для компоненти яскравості (і градації сірого кольору), іншу — для хроматичних компонент, є результатом тривалого дослідження з безліччю експериментів, проведених комітетом JPEG. Вони є частиною стандарту JPEG та відтворені в табл. Видно, як коефіцієнти QC таблиць зростають під час руху з лівого верхнього кута в правий нижній кут. У цьому відображається скорочення коефіцієнтів DCT, відповідних високим просторовим частотам.

16	11	10	16	24	40	51	61	17	18	24	47	99	99	99	99
12	12	14	19	26	58	60	55	18	21	26	66	99	99	99	99
14	13	16	24	40	57	69	56	24	26	56	99	99	99	99	99
14	17	22	29	51	87	80	62	47	66	99	99	99	99	99	99
18	22	37	56	68	109	103	77	99	99	99	99	99	99	99	99
24	35	55	64	81	104	113	92	99	99	99	99	99	99	99	99
49	64	78	87	103	121	120	101	99	99	99	99	99	99	99	99
72	92	95	98	112	100	103	99	99	99	99	99	99	99	99	99

Яскравість

Кольоровість

1. Таблиця квантування, прийнята за умовчанням. Дві такі таблиці, одна для компоненти яскравості (і градації сірого кольору), іншу — для хроматичних компонент, є результатом тривалого дослідження з безліччю експериментів, проведених комітетом JPEG. Вони є частиною стандарту JPEG та відтворені в табл. Видно, як коефіцієнти QC таблиць зростають під час руху з лівого верхнього кута в правий нижній кут. У цьому відображається скорочення коефіцієнтів DCT, відповідних високим просторовим частотам.

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Яскравість

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

Кольоровість

1. Таблиця квантування, прийнята за умовчанням. Дві такі таблиці, одна для компоненти яскравості (і градації сірого кольору), іншу — для хроматичних компонент, є результатом тривалого дослідження з безліччю експериментів, проведених комітетом JPEG. Вони є частиною стандарту JPEG та відтворені в табл. Видно, як коефіцієнти QC таблиць зростають під час руху з лівого верхнього кута в правий нижній кут. У цьому відображається скорочення коефіцієнтів DCT, відповідних високим просторовим частотам.

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Яскравість

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

Кольоровість

1. Таблиця квантування, прийнята за умовчанням. Дві такі таблиці, одна для компоненти яскравості (і градації сірого кольору), іншу — для хроматичних компонент, є результатом тривалого дослідження з безліччю експериментів, проведених комітетом JPEG. Вони є частиною стандарту JPEG та відтворені в табл. Видно, як коефіцієнти QC таблиць зростають під час руху з лівого верхнього кута в правий нижній кут. У цьому відображається скорочення коефіцієнтів DCT, відповідних високим просторовим частотам.

16	11	10	16	24	40	51	61	17	18	24	47	99	99	99	99
12	12	14	19	26	58	60	55	18	21	26	66	99	99	99	99
14	13	16	24	40	57	69	56	24	26	56	99	99	99	99	99
14	17	22	29	51	87	80	62	47	66	99	99	99	99	99	99
18	22	37	56	68	109	103	77	99	99	99	99	99	99	99	99
24	35	55	64	81	104	113	92	99	99	99	99	99	99	99	99
49	64	78	87	103	121	120	101	99	99	99	99	99	99	99	99
72	92	95	98	112	100	103	99	99	99	99	99	99	99	99	99

Яскравість

Кольоровість

1. Таблиця квантування, прийнята за умовчанням. Дві такі таблиці, одна для компоненти яскравості (і градації сірого кольору), іншу — для хроматичних компонент, є результатом тривалого дослідження з безліччю експериментів, проведених комітетом JPEG. Вони є частиною стандарту JPEG та відтворені в табл.

Видно, як коефіцієнти QC таблиць зростають під час руху з лівого верхнього кута в правий нижній кут. У цьому відображається скорочення коефіцієнтів DCT, відповідних високим просторовим частотам.

16	11	10	16	24	40	51	61	17	18	24	47	99	99	99	99
12	12	14	19	26	58	60	55	18	21	26	66	99	99	99	99
14	13	16	24	40	57	69	56	24	26	56	99	99	99	99	99
14	17	22	29	51	87	80	62	47	66	99	99	99	99	99	99
18	22	37	56	68	109	103	77	99	99	99	99	99	99	99	99
24	35	55	64	81	104	113	92	99	99	99	99	99	99	99	99
49	64	78	87	103	121	120	101	99	99	99	99	99	99	99	99
72	92	95	98	112	100	103	99	99	99	99	99	99	99	99	99

Яскравість

Кольоровість

1. Таблиця квантування, прийнята за умовчанням. Дві такі таблиці, одна для компоненти яскравості (і градації сірого кольору), іншу — для хроматичних компонент, є результатом тривалого дослідження з безліччю експериментів, проведених комітетом JPEG. Вони є частиною стандарту JPEG та відтворені в табл. Видно, як коефіцієнти QC таблиць зростають під час руху з лівого верхнього кута в правий нижній кут. У цьому відображається скорочення коефіцієнтів DCT, відповідних високим просторовим частотам.

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Яскравість

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

Кольоровість

1. Таблиця квантування, прийнята за умовчанням. Дві такі таблиці, одна для компоненти яскравості (і градації сірого кольору), іншу — для хроматичних компонент, є результатом тривалого дослідження з безліччю експериментів, проведених комітетом JPEG. Вони є частиною стандарту JPEG та відтворені в табл. Видно, як коефіцієнти QC таблиць зростають під час руху з лівого верхнього кута в правий нижній кут. У цьому відображається скорочення коефіцієнтів DCT, відповідних високим просторовим частотам.

2. Обчислюється проста таблиця коефіцієнтів квантування, яка залежить від параметра R , який задається користувачем. Прості вирази типу

$$Q_{ij} = 1 + (i + j) \cdot R$$

гарантують зменшення коефіцієнтів з лівого верхнього кута в правий нижній.

Якщо квантування зроблено правильно, то в блоці коефіцієнтів DCT залишиться лише кілька ненульових коефіцієнтів, які будуть сконцентровані у верхньому лівому куті матриці. Ці числа є виходом алгоритму JPEG, але слід ще стиснути перед записом у вихідний файл. У літературі з JPEG цей стиск називається “*ентропійним кодуванням*”, деталі якого будуть розбиратися в подальших лекціях. Три технічні прийоми використовуються при ентропійному кодуванні для стиснення цілочисельних матриць розміру 8×8 .

2. Обчислюється проста таблиця коефіцієнтів квантування, яка залежить від параметра R , який задається користувачем. Прості вирази типу

$$Q_{ij} = 1 + (i + j) \cdot R$$

гарантують зменшення коефіцієнтів з лівого верхнього кута в правий нижній.

Якщо квантування зроблено правильно, то в блоці коефіцієнтів DCT залишиться лише кілька ненульових коефіцієнтів, які будуть сконцентровані у верхньому лівому куті матриці. Ці числа є виходом алгоритму JPEG, але слід ще стиснути перед записом у вихідний файл. У літературі з JPEG цей стиск називається “*ентропійним кодуванням*”, деталі якого будуть розбиратися в подальших лекціях. Три технічні прийоми використовуються при ентропійному кодуванні для стиснення цілочисельних матриць розміру 8×8 .

2. Обчислюється проста таблиця коефіцієнтів квантування, яка залежить від параметра R , який задається користувачем. Прості вирази типу

$$Q_{ij} = 1 + (i + j) \cdot R$$

гарантують зменшення коефіцієнтів з лівого верхнього кута в правий нижній.

Якщо квантування зроблено правильно, то в блоці коефіцієнтів DCT залишиться лише кілька ненульових коефіцієнтів, які будуть сконцентровані у верхньому лівому куті матриці. Ці числа є виходом алгоритму JPEG, але слід ще стиснути перед записом у вихідний файл. У літературі з JPEG цей стиск називається “*ентропійним кодуванням*”, деталі якого будуть розбиратися в подальших лекціях. Три технічні прийоми використовуються при ентропійному кодуванні для стиснення цілочисельних матриць розміру 8×8 .

2. Обчислюється проста таблиця коефіцієнтів квантування, яка залежить від параметра R , який задається користувачем. Прості вирази типу

$$Q_{ij} = 1 + (i + j) \cdot R$$

гарантують зменшення коефіцієнтів з лівого верхнього кута в правий нижній.

Якщо квантування зроблено правильно, то в блоці коефіцієнтів DCT залишиться лише кілька ненульових коефіцієнтів, які будуть сконцентровані у верхньому лівому куті матриці. Ці числа є виходом алгоритму JPEG, але слід ще стиснути перед записом у вихідний файл. У літературі з JPEG цей стиск називається “*ентропійним кодуванням*”, деталі якого будуть розбиратися в подальших лекціях. Три технічні прийоми використовуються при ентропійному кодуванні для стиснення цілочисельних матриць розміру 8×8 .

2. Обчислюється проста таблиця коефіцієнтів квантування, яка залежить від параметра R , який задається користувачем. Прості вирази типу

$$Q_{ij} = 1 + (i + j) \cdot R$$

гарантують зменшення коефіцієнтів з лівого верхнього кута в правий нижній.

Якщо квантування зроблено правильно, то в блоці коефіцієнтів DCT залишаться лише кілька ненульових коефіцієнтів, які будуть сконцентровані у верхньому лівому куті матриці. Ці числа є виходом алгоритму JPEG, але слід ще стиснути перед записом у вихідний файл. У літературі з JPEG цей стиск називається “*ентропійним кодуванням*”, деталі якого будуть розбиратися в подальших лекціях. Три технічні прийоми використовуються при ентропійному кодуванні для стиснення цілочисельних матриць розміру 8×8 .

2. Обчислюється проста таблиця коефіцієнтів квантування, яка залежить від параметра R , який задається користувачем. Прості вирази типу

$$Q_{ij} = 1 + (i + j) \cdot R$$

гарантують зменшення коефіцієнтів з лівого верхнього кута в правий нижній.

Якщо квантування зроблено правильно, то в блоці коефіцієнтів DCT залишаться лише кілька ненульових коефіцієнтів, які будуть сконцентровані у верхньому лівому куті матриці. Ці числа є виходом алгоритму JPEG, але слід ще стиснути перед записом у вихідний файл. У літературі з JPEG цей стиск називається "*ентропійним кодуванням*", деталі якого будуть розбиратися в подальших лекціях. Три технічні прийоми використовуються при ентропійному кодуванні для стиснення цілочисельних матриць розміру 8×8 .

2. Обчислюється проста таблиця коефіцієнтів квантування, яка залежить від параметра R , який задається користувачем. Прості вирази типу

$$Q_{ij} = 1 + (i + j) \cdot R$$

гарантують зменшення коефіцієнтів з лівого верхнього кута в правий нижній.

Якщо квантування зроблено правильно, то в блоці коефіцієнтів DCT залишаться лише кілька ненульових коефіцієнтів, які будуть сконцентровані у верхньому лівому куті матриці. Ці числа є виходом алгоритму JPEG, але слід ще стиснути перед записом у вихідний файл. У літературі з JPEG цей стиск називається "*ентропійним кодуванням*", деталі якого будуть розбиратися в подальших лекціях. Три технічні прийоми використовуються при ентропійному кодуванні для стиснення цілочисельних матриць розміру 8×8 .

2. Обчислюється проста таблиця коефіцієнтів квантування, яка залежить від параметра R , який задається користувачем. Прості вирази типу

$$Q_{ij} = 1 + (i + j) \cdot R$$

гарантують зменшення коефіцієнтів з лівого верхнього кута в правий нижній.

Якщо квантування зроблено правильно, то в блоці коефіцієнтів DCT залишаться лише кілька ненульових коефіцієнтів, які будуть сконцентровані у верхньому лівому куті матриці. Ці числа є виходом алгоритму JPEG, але слід ще стиснути перед записом у вихідний файл. У літературі з JPEG цей стиск називається “*ентропійним кодуванням*”, деталі якого будуть розбиратися в подальших лекціях. Три технічні прийоми використовуються при ентропійному кодуванні для стиснення цілочисельних матриць розміру 8×8 .

2. Обчислюється проста таблиця коефіцієнтів квантування, яка залежить від параметра R , який задається користувачем. Прості вирази типу

$$Q_{ij} = 1 + (i + j) \cdot R$$

гарантують зменшення коефіцієнтів з лівого верхнього кута в правий нижній.

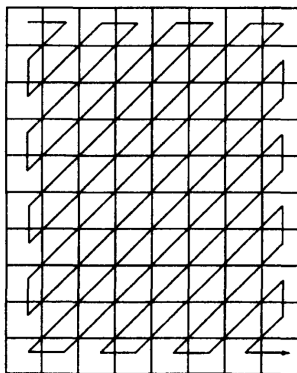
Якщо квантування зроблено правильно, то в блоці коефіцієнтів DCT залишиться лише кілька ненульових коефіцієнтів, які будуть сконцентровані у верхньому лівому куті матриці. Ці числа є виходом алгоритму JPEG, але слід ще стиснути перед записом у вихідний файл. У літературі з JPEG цей стиск називається “*ентропійним кодуванням*”, деталі якого будуть розбиратися в подальших лекціях. Три технічні прийоми використовуються при ентропійному кодуванні для стиснення цілочисельних матриць розміру 8×8 .

2. Обчислюється проста таблиця коефіцієнтів квантування, яка залежить від параметра R , який задається користувачем. Прості вирази типу

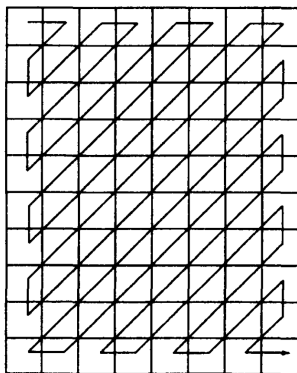
$$Q_{ij} = 1 + (i + j) \cdot R$$

гарантують зменшення коефіцієнтів з лівого верхнього кута в правий нижній.

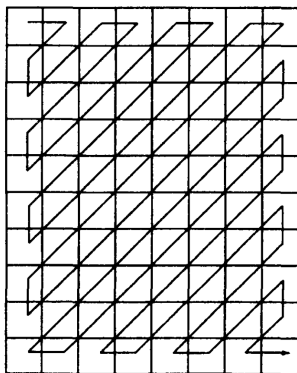
Якщо квантування зроблено правильно, то в блоці коефіцієнтів DCT залишиться лише кілька ненульових коефіцієнтів, які будуть сконцентровані у верхньому лівому куті матриці. Ці числа є виходом алгоритму JPEG, але слід ще стиснути перед записом у вихідний файл. У літературі з JPEG цей стиск називається “*ентропійним кодуванням*”, деталі якого будуть розбиратися в подальших лекціях. Три технічні прийоми використовуються при ентропійному кодуванні для стиснення цілочисельних матриць розміру 8×8 .



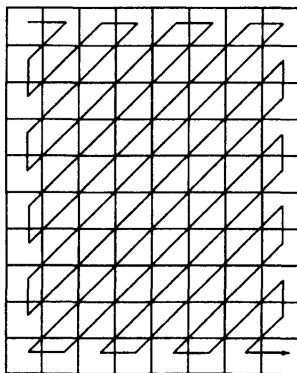
1. 64 числа вишиковуються одне за одним як при скануванні зигзагом (див. рис.). На початку стоять ненульові числа, за якими зазвичай слідує довгий хвіст з одних нулів. У файл виводяться лише ненульові числа (після належного кодування), за якими слідує спеціальний код EOB (end-of-block, кінець блоку). Немає потреби записувати весь хвіст нулів (можна також сказати, що EOB кодує довгу серію нулів).



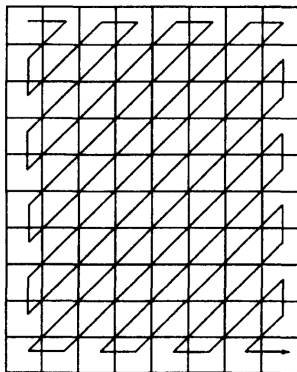
1. 64 числа вишиковуються одне за одним як при скануванні зигзагом (див. рис.). На початку стоять ненульові числа, за якими зазвичай слідує довгий хвіст з одних нулів. У файл виводяться лише ненульові числа (після належного кодування), за якими слідує спеціальний код EOB (end-of-block, кінець блоку). Немає потреби записувати весь хвіст нулів (можна також сказати, що EOB кодує довгу серію нулів).



1. 64 числа вишиковуються одне за одним як при скануванні зигзагом (див. рис.). На початку стоять ненульові числа, за якими зазвичай слідує довгий хвіст з одних нулів. У файл виводяться лише ненульові числа (після належного кодування), за якими слідує спеціальний код EOB (end-of-block, кінець блоку). Немає потреби записувати весь хвіст нулів (можна також сказати, що EOB кодує довгу серію нулів).



1. 64 числа вишиковуються одне за одним як при скануванні зигзагом (див. рис.). На початку стоять ненульові числа, за якими зазвичай слідує довгий хвіст з одних нулів. У файл виводяться лише ненульові числа (після належного кодування), за якими слідує спеціальний код EOB (end-of-block, кінець блоку). Немає потреби записувати весь хвіст нулів (можна також сказати, що EOB кодує довгу серію нулів).



1. 64 числа вишиковуються одне за одним як при скануванні зигзагом (див. рис.). На початку стоять ненульові числа, за якими зазвичай слідує довгий хвіст з одних нулів. У файл виводяться лише ненульові числа (після належного кодування), за якими слідує спеціальний код EOB (end-of-block, кінець блоку). Немає потреби записувати весь хвіст нулів (можна також сказати, що EOB кодує довгу серію нулів).

1118	2	0	0	0	0	0	0
0	0	0	0	0	0	0	0
-2	0	0	-1	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Приклад

У табл. наведено список гіпотетичних коефіцієнтів DCT, з яких лише 4 не дорівнюють нулю. При зигзагоподібному упорядкуванні цих чисел виходить послідовність коефіцієнтів:

$$1118, 2, 3, \underbrace{-2, 0, \dots, 0}_{13}, \underbrace{-1, 0, \dots, 0}_{46}$$

1118	2	0	0	0	0	0	0
0	0	0	0	0	0	0	0
-2	0	0	-1	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Приклад

У табл. наведено список гіпотетичних коефіцієнтів DCT, з яких лише 4 не дорівнюють нулю. При зигзагоподібному упорядкуванні цих чисел виходить послідовність коефіцієнтів:

$$1118, 2, 3, \underbrace{-2, 0, \dots, 0}_{13}, \underbrace{-1, 0, \dots, 0}_{46}.$$

1118	2	0	0	0	0	0	0
0	0	0	0	0	0	0	0
-2	0	0	-1	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Приклад

У табл. наведено список гіпотетичних коефіцієнтів DCT, з яких лише 4 не дорівнюють нулю. При зигзагоподібному упорядкуванні цих чисел виходить послідовність коефіцієнтів:

$$1118, 2, 3, \underbrace{-2, 0, \dots, 0}_{13}, \underbrace{-1, 0, \dots, 0}_{46}.$$

1118	2	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
-2	0	0	-1	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

Приклад

У табл. наведено список гіпотетичних коефіцієнтів DCT, з яких лише 4 не дорівнюють нулю. При зигзагоподібному упорядкуванні цих чисел виходить послідовність коефіцієнтів:

$$1118, 2, 3, \underbrace{-2, 0, \dots, 0}_{13}, \underbrace{-1, 0, \dots, 0}_{46}.$$

1118	2	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
-2	0	0	-1	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

Приклад

У табл. наведено список гіпотетичних коефіцієнтів DCT, з яких лише 4 не дорівнюють нулю. При зигзагоподібному упорядкуванні цих чисел виходить послідовність коефіцієнтів:

$$1118, 2, 3, \underbrace{-2, 0, \dots, 0}_{13}, \underbrace{-1, 0, \dots, 0}_{46}.$$

1118	2	0	0	0	0	0	0
0	0	0	0	0	0	0	0
-2	0	0	-1	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Приклад

У табл. наведено список гіпотетичних коефіцієнтів DCT, з яких лише 4 не дорівнюють нулю. При зигзагоподібному упорядкуванні цих чисел виходить послідовність коефіцієнтів:

$$1118, 2, 3, \underbrace{-2, 0, \dots, 0}_{13}, -1, \underbrace{0, \dots, 0}_{46}.$$

Стиснення зображень. JPEG. Квантування

(0, 0)	(0, 1)	(1, 0)	(2, 0)	(1, 1)	(0, 2)	(0, 3)	(1, 2)
(2, 1)	(3, 0)	(4, 0)	(3, 1)	(2, 2)	(1, 3)	(0, 4)	(0, 5)
(1, 4)	(2, 3)	(3, 2)	(4, 1)	(5, 0)	(6, 0)	(5, 1)	(4, 2)
(3, 3)	(2, 4)	(1, 5)	(0, 6)	(0, 7)	(1, 6)	(2, 5)	(3, 4)
(4, 3)	(5, 2)	(6, 1)	(7, 0)	(7, 1)	(6, 2)	(5, 3)	(4, 4)
(3, 5)	(2, 6)	(1, 7)	(2, 7)	(3, 6)	(4, 5)	(5, 4)	(6, 3)
(7, 2)	(7, 3)	(6, 4)	(5, 5)	(4, 6)	(3, 7)	(4, 7)	(5, 6)
(6, 5)	(7, 4)	(7, 5)	(6, 6)	(5, 7)	(6, 7)	(7, 6)	(7, 7)

А як написати підпрограму для зчитування елементів матриці по зигзагу? Найпростіший спосіб полягає в ручному простеженні цього шляху і запису результату в масив структур `zz`, в якому кожна структура складається з пари координат клітин, через які проходить зигзагоподібний шлях (див. рис.)

Якщо компоненти структури `zz` позначити `zz.r` та `zz.c`, то шлях по зигзагу можна зробити за допомогою такого циклу

```
for (i=0; i<64; i++){  
    row:=zz[i].r; col:=zz [i].c  
    ...data.imit[row] [col]...}
```

2. Ненульові коефіцієнти перетворення стискаються методом Гаффмана.
3. Перше з цих чисел (коефіцієнт DC) обробляється окремо від інших чисел (коефіцієнтів AC).

Стиснення зображень. JPEG. Квантування

(0, 0)	(0, 1)	(1, 0)	(2, 0)	(1, 1)	(0, 2)	(0, 3)	(1, 2)
(2, 1)	(3, 0)	(4, 0)	(3, 1)	(2, 2)	(1, 3)	(0, 4)	(0, 5)
(1, 4)	(2, 3)	(3, 2)	(4, 1)	(5, 0)	(6, 0)	(5, 1)	(4, 2)
(3, 3)	(2, 4)	(1, 5)	(0, 6)	(0, 7)	(1, 6)	(2, 5)	(3, 4)
(4, 3)	(5, 2)	(6, 1)	(7, 0)	(7, 1)	(6, 2)	(5, 3)	(4, 4)
(3, 5)	(2, 6)	(1, 7)	(2, 7)	(3, 6)	(4, 5)	(5, 4)	(6, 3)
(7, 2)	(7, 3)	(6, 4)	(5, 5)	(4, 6)	(3, 7)	(4, 7)	(5, 6)
(6, 5)	(7, 4)	(7, 5)	(6, 6)	(5, 7)	(6, 7)	(7, 6)	(7, 7)

А як написати підпрограму для зчитування елементів матриці по зигзагу?

Найпростіший спосіб полягає в ручному простеженні цього шляху і запису результату в масив структур `zz`, в якому кожна структура складається з пари координат клітин, через які проходить зигзагоподібний шлях (див. рис.)

Якщо компоненти структури `zz` позначити `zz.r` та `zz.c`, то шлях по зигзагу можна зробити за допомогою такого циклу

```
for (i=0; i<64; i++){  
    row:=zz[i].r; col:=zz [i].c  
    ...data.imit[row] [col]...}
```

2. Ненульові коефіцієнти перетворення стискаються методом Гаффмана.
3. Перше з цих чисел (коефіцієнт DC) обробляється окремо від інших чисел (коефіцієнтів AC).

Стиснення зображень. JPEG. Квантування

(0, 0)	(0, 1)	(1, 0)	(2, 0)	(1, 1)	(0, 2)	(0, 3)	(1, 2)
(2, 1)	(3, 0)	(4, 0)	(3, 1)	(2, 2)	(1, 3)	(0, 4)	(0, 5)
(1, 4)	(2, 3)	(3, 2)	(4, 1)	(5, 0)	(6, 0)	(5, 1)	(4, 2)
(3, 3)	(2, 4)	(1, 5)	(0, 6)	(0, 7)	(1, 6)	(2, 5)	(3, 4)
(4, 3)	(5, 2)	(6, 1)	(7, 0)	(7, 1)	(6, 2)	(5, 3)	(4, 4)
(3, 5)	(2, 6)	(1, 7)	(2, 7)	(3, 6)	(4, 5)	(5, 4)	(6, 3)
(7, 2)	(7, 3)	(6, 4)	(5, 5)	(4, 6)	(3, 7)	(4, 7)	(5, 6)
(6, 5)	(7, 4)	(7, 5)	(6, 6)	(5, 7)	(6, 7)	(7, 6)	(7, 7)

А як написати підпрограму для зчитування елементів матриці по зигзагу? Найпростіший спосіб полягає в ручному простеженні цього шляху і запису результату в масив структур `zz`, в якому кожна структура складається з пари координат клітин, через які проходить зигзагоподібний шлях (див. рис.)

Якщо компоненти структури `zz` позначити `zz.r` та `zz.c`, то шлях по зигзагу можна зробити за допомогою такого циклу

```
for (i=0; i<64; i++){  
    row:=zz[i].r; col:=zz [i].c  
    ...data.imit[row] [col]...}
```

2. Ненульові коефіцієнти перетворення стискаються методом Гаффмана.
3. Перше з цих чисел (коефіцієнт DC) обробляється окремо від інших чисел (коефіцієнтів AC).

Стиснення зображень. JPEG. Квантування

(0, 0)	(0, 1)	(1, 0)	(2, 0)	(1, 1)	(0, 2)	(0, 3)	(1, 2)
(2, 1)	(3, 0)	(4, 0)	(3, 1)	(2, 2)	(1, 3)	(0, 4)	(0, 5)
(1, 4)	(2, 3)	(3, 2)	(4, 1)	(5, 0)	(6, 0)	(5, 1)	(4, 2)
(3, 3)	(2, 4)	(1, 5)	(0, 6)	(0, 7)	(1, 6)	(2, 5)	(3, 4)
(4, 3)	(5, 2)	(6, 1)	(7, 0)	(7, 1)	(6, 2)	(5, 3)	(4, 4)
(3, 5)	(2, 6)	(1, 7)	(2, 7)	(3, 6)	(4, 5)	(5, 4)	(6, 3)
(7, 2)	(7, 3)	(6, 4)	(5, 5)	(4, 6)	(3, 7)	(4, 7)	(5, 6)
(6, 5)	(7, 4)	(7, 5)	(6, 6)	(5, 7)	(6, 7)	(7, 6)	(7, 7)

А як написати підпрограму для зчитування елементів матриці по зигзагу? Найпростіший спосіб полягає в ручному простеженні цього шляху і запису результату в масив структур `zz`, в якому кожна структура складається з пари координат клітин, через які проходить зигзагоподібний шлях (див. рис.)

Якщо компоненти структури `zz` позначити `zz.r` та `zz.c`, то шлях по зигзагу можна зробити за допомогою такого циклу

```
for (i=0; i<64; i++){  
    row:=zz[i].r; col:=zz [i].c  
    ...data.imit[row] [col]...}
```

2. Ненульові коефіцієнти перетворення стискаються методом Гаффмана.
3. Перше з цих чисел (коефіцієнт DC) обробляється окремо від інших чисел (коефіцієнтів AC).

Стиснення зображень. JPEG. Квантування

(0, 0)	(0, 1)	(1, 0)	(2, 0)	(1, 1)	(0, 2)	(0, 3)	(1, 2)
(2, 1)	(3, 0)	(4, 0)	(3, 1)	(2, 2)	(1, 3)	(0, 4)	(0, 5)
(1, 4)	(2, 3)	(3, 2)	(4, 1)	(5, 0)	(6, 0)	(5, 1)	(4, 2)
(3, 3)	(2, 4)	(1, 5)	(0, 6)	(0, 7)	(1, 6)	(2, 5)	(3, 4)
(4, 3)	(5, 2)	(6, 1)	(7, 0)	(7, 1)	(6, 2)	(5, 3)	(4, 4)
(3, 5)	(2, 6)	(1, 7)	(2, 7)	(3, 6)	(4, 5)	(5, 4)	(6, 3)
(7, 2)	(7, 3)	(6, 4)	(5, 5)	(4, 6)	(3, 7)	(4, 7)	(5, 6)
(6, 5)	(7, 4)	(7, 5)	(6, 6)	(5, 7)	(6, 7)	(7, 6)	(7, 7)

А як написати підпрограму для зчитування елементів матриці по зигзагу? Найпростіший спосіб полягає в ручному простеженні цього шляху і запису результату в масив структур `zz`, в якому кожна структура складається з пари координат клітин, через які проходить зигзагоподібний шлях (див. рис.)

Якщо компоненти структури `zz` позначити `zz.r` та `zz.c`, то шлях по зигзагу можна зробити за допомогою такого циклу

```
for (i=0; i<64; i++){  
    row:=zz[i].r; col:=zz [i].c  
    ...data.imit[row] [col]...}
```

2. Ненульові коефіцієнти перетворення стискаються методом Гаффмана.
3. Перше з цих чисел (коефіцієнт DC) обробляється окремо від інших чисел (коефіцієнтів AC).

Стиснення зображень. JPEG. Квантування

(0, 0)	(0, 1)	(1, 0)	(2, 0)	(1, 1)	(0, 2)	(0, 3)	(1, 2)
(2, 1)	(3, 0)	(4, 0)	(3, 1)	(2, 2)	(1, 3)	(0, 4)	(0, 5)
(1, 4)	(2, 3)	(3, 2)	(4, 1)	(5, 0)	(6, 0)	(5, 1)	(4, 2)
(3, 3)	(2, 4)	(1, 5)	(0, 6)	(0, 7)	(1, 6)	(2, 5)	(3, 4)
(4, 3)	(5, 2)	(6, 1)	(7, 0)	(7, 1)	(6, 2)	(5, 3)	(4, 4)
(3, 5)	(2, 6)	(1, 7)	(2, 7)	(3, 6)	(4, 5)	(5, 4)	(6, 3)
(7, 2)	(7, 3)	(6, 4)	(5, 5)	(4, 6)	(3, 7)	(4, 7)	(5, 6)
(6, 5)	(7, 4)	(7, 5)	(6, 6)	(5, 7)	(6, 7)	(7, 6)	(7, 7)

А як написати підпрограму для зчитування елементів матриці по зигзагу? Найпростіший спосіб полягає в ручному простеженні цього шляху і запису результату в масив структур `zz`, в якому кожна структура складається з пари координат клітин, через які проходить зигзагоподібний шлях (див. рис.)

Якщо компоненти структури `zz` позначити `zz.r` та `zz.c`, то шлях по зигзагу можна зробити за допомогою такого циклу

```
for (i=0; i<64; i++){  
    row:=zz[i].r; col:=zz [i].c  
    ...data.imit[row] [col]...}
```

2. Ненульові коефіцієнти перетворення стискаються методом Гаффмана.
3. Перше з цих чисел (коефіцієнт DC) обробляється окремо від інших чисел (коефіцієнтів AC).

Стиснення зображень. JPEG. Квантування

(0, 0)	(0, 1)	(1, 0)	(2, 0)	(1, 1)	(0, 2)	(0, 3)	(1, 2)
(2, 1)	(3, 0)	(4, 0)	(3, 1)	(2, 2)	(1, 3)	(0, 4)	(0, 5)
(1, 4)	(2, 3)	(3, 2)	(4, 1)	(5, 0)	(6, 0)	(5, 1)	(4, 2)
(3, 3)	(2, 4)	(1, 5)	(0, 6)	(0, 7)	(1, 6)	(2, 5)	(3, 4)
(4, 3)	(5, 2)	(6, 1)	(7, 0)	(7, 1)	(6, 2)	(5, 3)	(4, 4)
(3, 5)	(2, 6)	(1, 7)	(2, 7)	(3, 6)	(4, 5)	(5, 4)	(6, 3)
(7, 2)	(7, 3)	(6, 4)	(5, 5)	(4, 6)	(3, 7)	(4, 7)	(5, 6)
(6, 5)	(7, 4)	(7, 5)	(6, 6)	(5, 7)	(6, 7)	(7, 6)	(7, 7)

А як написати підпрограму для зчитування елементів матриці по зигзагу? Найпростіший спосіб полягає в ручному простеженні цього шляху і запису результату в масив структур `zz`, в якому кожна структура складається з пари координат клітин, через які проходить зигзагоподібний шлях (див. рис.)

Якщо компоненти структури `zz` позначити `zz.r` та `zz.c`, то шлях по зигзагу можна зробити за допомогою такого циклу

```
for (i=0; i<64; i++){  
    row:=zz[i].r; col:=zz [i].c  
    ...data.imit[row] [col]...}
```

2. Ненульові коефіцієнти перетворення стискаються методом Гаффмана.
3. Перше з цих чисел (коефіцієнт DC) обробляється окремо від інших чисел (коефіцієнтів AC).

Стиснення зображень. JPEG. Квантування

(0, 0)	(0, 1)	(1, 0)	(2, 0)	(1, 1)	(0, 2)	(0, 3)	(1, 2)
(2, 1)	(3, 0)	(4, 0)	(3, 1)	(2, 2)	(1, 3)	(0, 4)	(0, 5)
(1, 4)	(2, 3)	(3, 2)	(4, 1)	(5, 0)	(6, 0)	(5, 1)	(4, 2)
(3, 3)	(2, 4)	(1, 5)	(0, 6)	(0, 7)	(1, 6)	(2, 5)	(3, 4)
(4, 3)	(5, 2)	(6, 1)	(7, 0)	(7, 1)	(6, 2)	(5, 3)	(4, 4)
(3, 5)	(2, 6)	(1, 7)	(2, 7)	(3, 6)	(4, 5)	(5, 4)	(6, 3)
(7, 2)	(7, 3)	(6, 4)	(5, 5)	(4, 6)	(3, 7)	(4, 7)	(5, 6)
(6, 5)	(7, 4)	(7, 5)	(6, 6)	(5, 7)	(6, 7)	(7, 6)	(7, 7)

А як написати підпрограму для зчитування елементів матриці по зигзагу? Найпростіший спосіб полягає в ручному простеженні цього шляху і запису результату в масив структур `zz`, в якому кожна структура складається з пари координат клітин, через які проходить зигзагоподібний шлях (див. рис.)

Якщо компоненти структури `zz` позначити `zz.r` та `zz.c`, то шлях по зигзагу можна зробити за допомогою такого циклу

```
for (i=0; i<64; i++){  
    row:=zz[i].r; col:=zz [i].c  
    ...data.imit[row] [col]...}
```

2. Ненульові коефіцієнти перетворення стискаються методом Гаффмана.
3. Перше з цих чисел (коефіцієнт DC) обробляється окремо від інших чисел (коефіцієнтів AC).

Стиснення зображень. JPEG. Квантування

(0, 0)	(0, 1)	(1, 0)	(2, 0)	(1, 1)	(0, 2)	(0, 3)	(1, 2)
(2, 1)	(3, 0)	(4, 0)	(3, 1)	(2, 2)	(1, 3)	(0, 4)	(0, 5)
(1, 4)	(2, 3)	(3, 2)	(4, 1)	(5, 0)	(6, 0)	(5, 1)	(4, 2)
(3, 3)	(2, 4)	(1, 5)	(0, 6)	(0, 7)	(1, 6)	(2, 5)	(3, 4)
(4, 3)	(5, 2)	(6, 1)	(7, 0)	(7, 1)	(6, 2)	(5, 3)	(4, 4)
(3, 5)	(2, 6)	(1, 7)	(2, 7)	(3, 6)	(4, 5)	(5, 4)	(6, 3)
(7, 2)	(7, 3)	(6, 4)	(5, 5)	(4, 6)	(3, 7)	(4, 7)	(5, 6)
(6, 5)	(7, 4)	(7, 5)	(6, 6)	(5, 7)	(6, 7)	(7, 6)	(7, 7)

А як написати підпрограму для зчитування елементів матриці по зигзагу? Найпростіший спосіб полягає в ручному простеженні цього шляху і запису результату в масив структур `zz`, в якому кожна структура складається з пари координат клітин, через які проходить зигзагоподібний шлях (див. рис.)

Якщо компоненти структури `zz` позначити `zz.r` та `zz.c`, то шлях по зигзагу можна зробити за допомогою такого циклу

```
for (i=0; i<64; i++){  
    row:=zz[i].r; col:=zz [i].c  
    ...data.imit[row] [col]...}
```

2. Ненульові коефіцієнти перетворення стискаються методом Гаффмана.
3. Перше з цих чисел (коефіцієнт DC) обробляється окремо від інших чисел (коефіцієнтів AC).

Дякую за увагу!