**XXX INTERNATIONAL SCIENTIFIC STUDENT CONFERENCE**

**SAKHAROV READINGS**

**THE ALTERNATIVE SOLUTIONS OF COMPLEX LIGHTS OUT PUZZLES**

Rattanakarm S., Yoosiri B., Pukdee J., Amornvatcharapong A.

*Mahidol Wittayanusorn School, Nakhon Pathom, Thailand*

*sitchai0645@gmail.com*

The Alternative Solutions of Complex Lights Out Puzzles project has the purpose to develop the classic Lights Out Puzzle research. Lights Out Puzzle is a light-off-bulb puzzle which the aim of the game is to turn off all the bulbs. The Lights Out Puzzle Research is about the answer of 3x3 square table with 2 states: On (1) and Off (0). The game will start with square table. In each grid of the table is a light bulb with on or off state. Players are able to change the status of each light bulb by pressing on it with the conditions that the pressed light bulb and the light bulb that shares the common side with the pressed light bulb will be changed as well. Hence, the game will end when all light bulbs are on off states. This makes it a form of project that leads to any  grid layout with any states on any surface, whereas both the pressed light bulb and the corresponding light bulbs will be changed. The objective is to generalize the Lights Out Puzzle Research and find out how to find the answer with the shortest step. In this regard, we started by using the information from the research helping for summarizing the relationships and the definition of the knowledge needed in this project. Then, we used mathematical knowledge to prove the theorem and to find the necessary answers for this project. After that, we used that information and the observation skills to separate the cases. We also coded the program to examine our answer-finding methods for the confirmation that these methods are practical, for the facilitation of the experiment, and for the understanding of the people who interested in this project.

Next is the summarized procedure of the project. At first, we will only consider the square table with  states. Since we can find the answer of the initial table (the table we want to find answer) if and only if we can find the way to press the all-off-light-bulb table to that table. In an table, we called each grid of the table as the i-th-box where i is the place number of the grid on the table. We define  as the  matrix that is the state of i-th-box when we only press the all-off-light-bulb table at the i-th-box. Therefore, we get  where  is the number of times we press on the i-th-box and  is the initial table. Then, we can transform this equation by using linear algebra into  where  is the  matrix that has all of the members in the i-th-column as the member of  and  where  is the number of times we press on the i-th-box and . As the table has  states that is to say the states can be only , we will get . From the equation,  is the way to press from where 0 is all-off-light-bulb table and  is the initial table. So,  will be the answer that is the way to press from . Because there are some tables we can’t find the answer or the way to press from , we will find the way to press from where  is the d-state-table which all light bulbs are on d state, , and  is the state of the initial table. Generally, the answers will be finite or infinite. As we consider in modulo , the answers should be finite. By calculating, the number of  table with  states that can find the answer is  and the number of the table that can’t find the answer is  where  is the number of the answers.

Lemma  is constant in an  table with  states.

Pf. Let  is the set of all ways to press from  for some integers  , let  be the way to press from  for some table . we will get the set  where each of the members is the way to press from . Suppose that, there is  which is the way to press from  and . Hence, we get  being the way to press from . Thus, . Since, so  a contradiction. Therefore,  is constant in an  table with  states. 

In the next step, we will consider the table that can find the answer. From  and the matrix  and  that we know their value, we will consider  that is in the augmented matrix form. Thus, we can find  by multiplying . We will get . The members of  could be the fraction that we did not define because. And, sometimes we can’t find  because . So, from  will we reduce  where  is the row echelon form. Therefrom, we will change  to the identity matrix () by using the lemma. After we reduce , we will get  while  will be the value of the i-th-roll of .

Lemma For  , then  where   such that .

Pf. Let  be a positive integer that  and .

From Bezout’s identity we know that  where ,  that . Therefore, . We call  as a multiplication inverse of  modulo .

Due to the reducing from , we know that the product of the value that is on diagonal line of  is equal to . From the lemma, we could find  which  for every  that lie on the diagonal line of  while . Thus, we can convert  and we will get . For this reason, we can always find the answer of the initial table when  where  is the state of initial table and .

From the above, it is only one case that the initial table has the answer. We know that there are some tables that we can’t find the  or when . Allowing us to divide the cases for the consideration as follows:

1. 

1.1  and 

1.2  and 

2. 

For all cases, we will consider in an  table with  states starting from  while  will be the value of the i-th-roll of  and we call the value of the number that lie on the diagonal of  on i-th-roll as .

Case 1.1  and 

We have already finished case 2.1 from the above.

Case 1.2  and 

When we reduce , we can interpret that there are at least one i such that . So, we can’t find the multiplication inverse of  modulo .

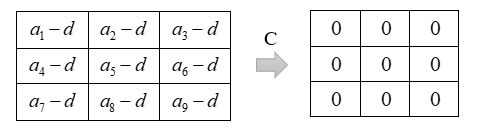
Hence, we divide into 2 cases namely:

1.  This case is similar to Case , so we will talk later.

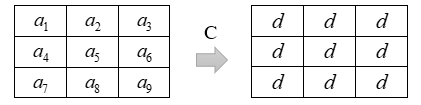
2.  We can divide this case into 2 sub-cases that is:

2.1  For this case we can do the similar things to the Case 2.1  and , then we can find the answer easily.

2.2  This case show that we can’t find the answer of the initial table. So, we will change the initial table to be the d-state table while  and  is the state of initial table. At first, we suppose   . Assume that, we can change the initial table to the d-state table. So, we will get new  such  and we will get the relationship between  and  that is  for some integer . We can summarize as in the picture.



Therefore



And we will find the answer (C), if there is one.

Case 2  When we reduce , we can interpret that there are at least one i such that . For the other , , we have 2 cases that are  and . So, we do the same things as the above. Since we have some i that , the number of the equations less than the number of the variables. Thus, we have the free variable. For this reason, we will have many answers and the number of the answers is  while m is the state of the initial table and j is the number of free variables. After that, we will divide to 2 cases as follows:

1.  Therefore,  makes the equation true and we can find the answer by considering the other ,  the same way as the above.

2.  Therefore,  is not accordance with the equation. Then, we change the initial table to be the d-state table while  as the above.

All of these are methods of finding the answer in any square tables and any states while the purpose of this project is to find the answer in any  grid layout with any states on any surface whereas both the pressed light bulb and the corresponding light bulbs will be changed. Actually, we use the same method to find the answer. We only have to change  because it is defined in a relation with pressing and the size of table. So, the pressing that effect the light bulbs in other ways will change  and the size of the table will change the size of  too. The developers have already coded the program to testify the methods and this program can also find the answers. This knowledge can use in daily life for example: 1) watering the plants since in the present-day people install sprinkler regardless as the unequal volume of each plant needs. So, we convert the garden to table and replace the units of water that the plants need in each box of the table. After that, we use the knowledge from this project to install the sprinkler. So, the plants will get the volume of water and the actually need. It helps us to save money and save water. 2) Detecting sensors as nowadays there are some sensors that have a high chance of error which is harmful and ineffective. Hence, we will help to increase the efficiency by converting the room to the table, setting the area that can be detected by one sensor. Then, we will use the knowledge from this project to install the sensor which each area can be detected by 3 sensors and let the sensors vote if that area has people or not. We hope that this knowledge could help inventing something new and benefit the world.

[1]Eisele, R. (2018). LightsOut Solution using Linear Algebra. Retrieved from https://www.xarg.org/2018/07/lightsout-solution-using-linear-algebra/

[2]Weisstein, E. W. (n.d.-a). Lights Out Puzzle. *MathWorld*. Retrieved from <http://mathworld.wolfram.com/LightsOutPuzzle.html>

**XXX МЕЖДУНАРОДНАЯ НАУЧНАЯ КОНФЕРЕНЦИЯ ШКОЛЬНИКОВ**

**САХАРОВСКИЕ ЧТЕНИЯ**

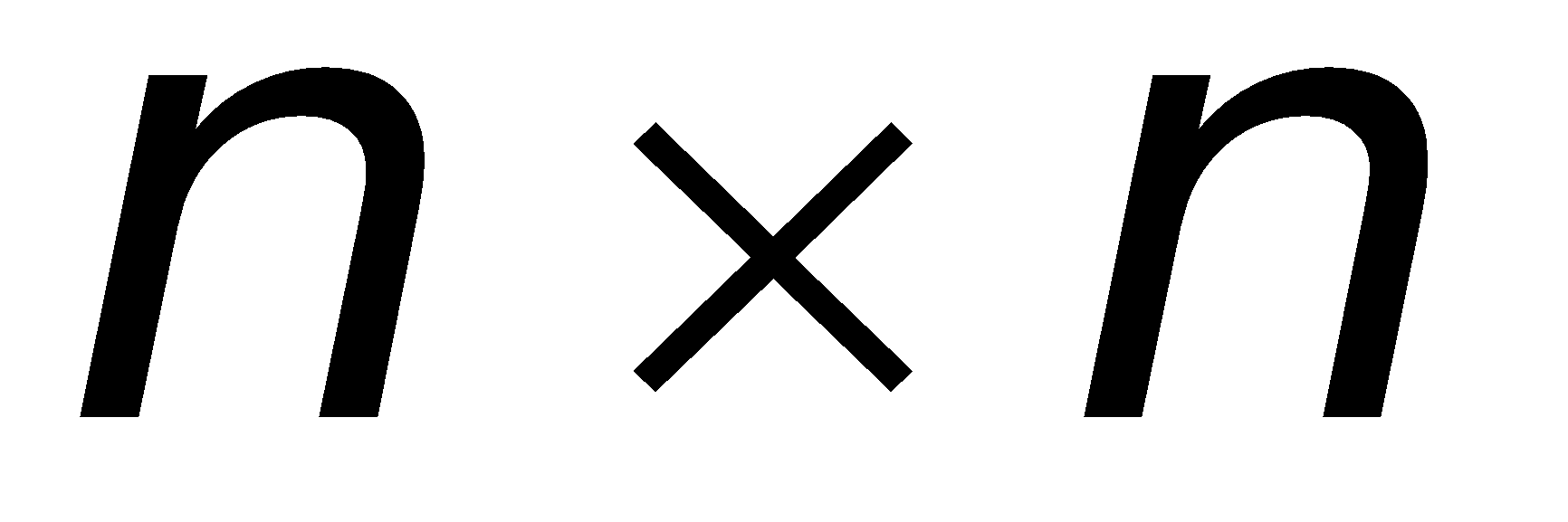
**АЛЬТЕРНАТИВНЫЕ РЕШЕНИЯ СЛОЖНОГО ПАЗЛА LIGHTS OUT**

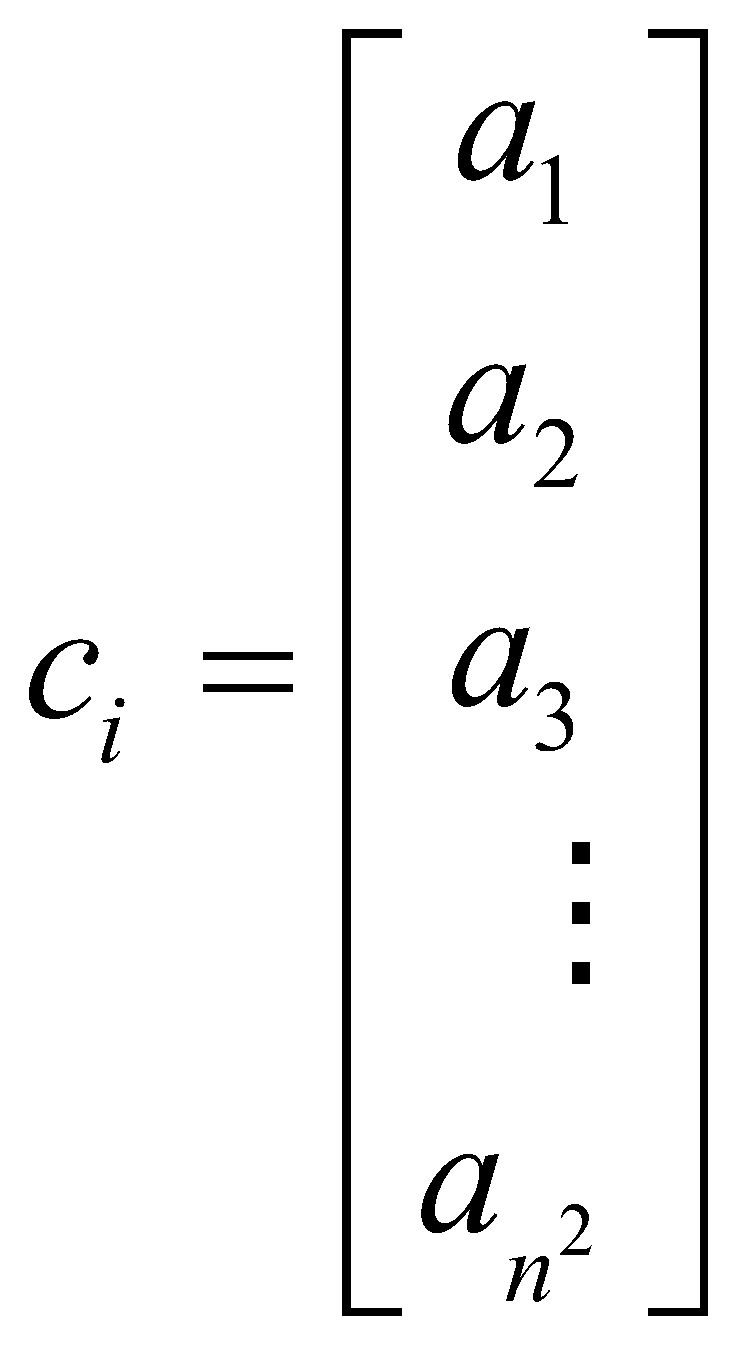
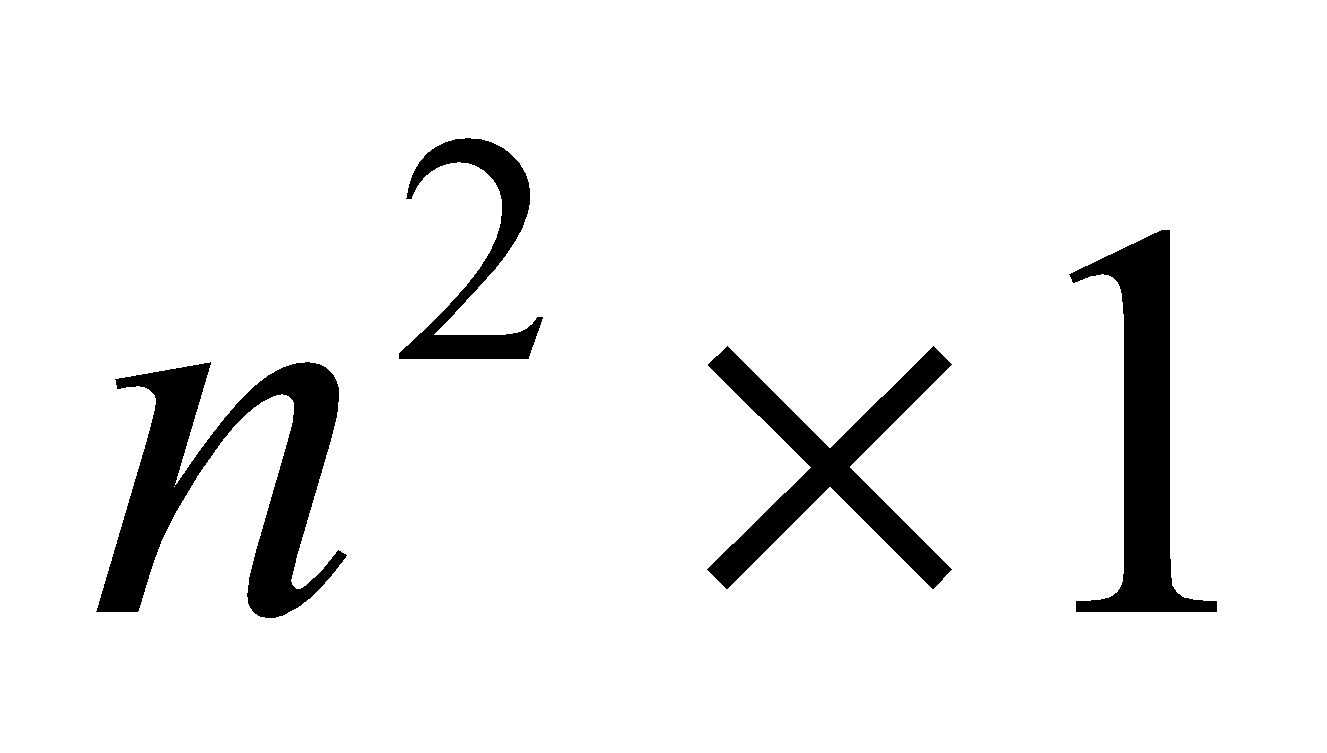
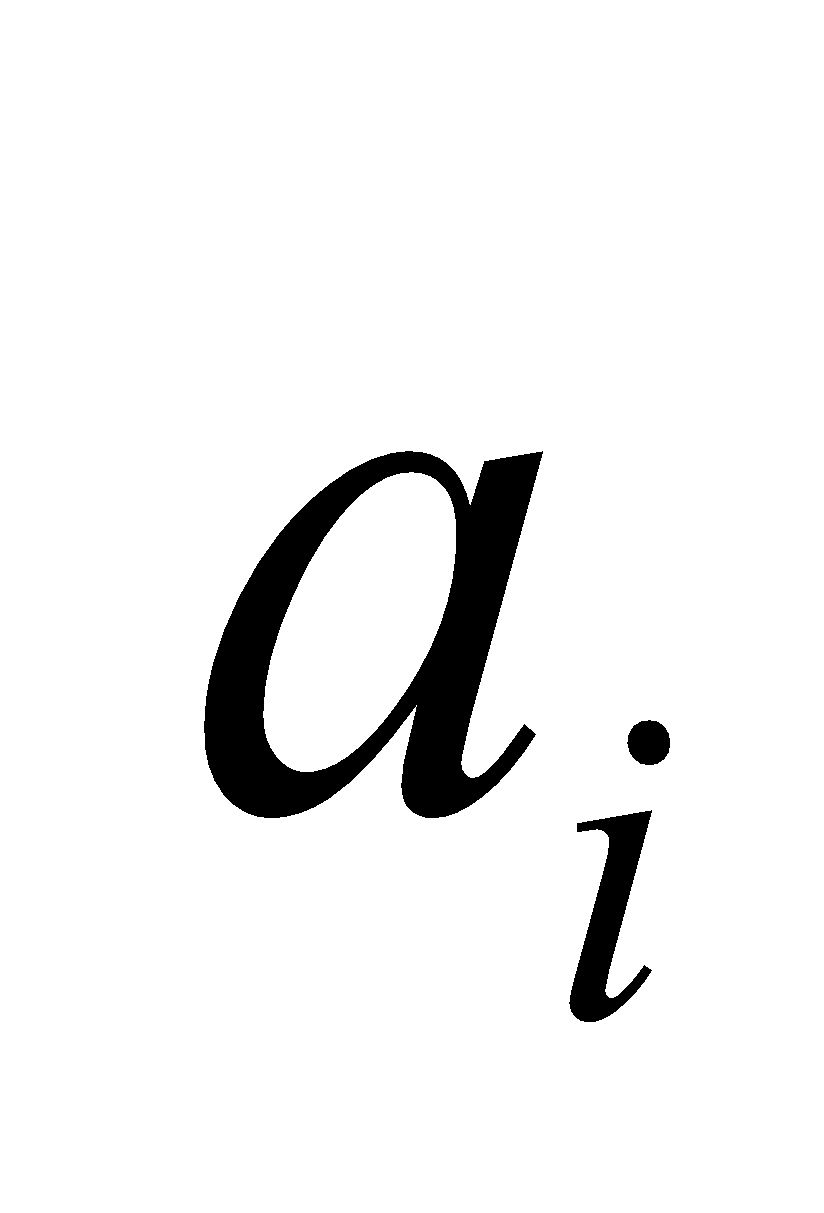
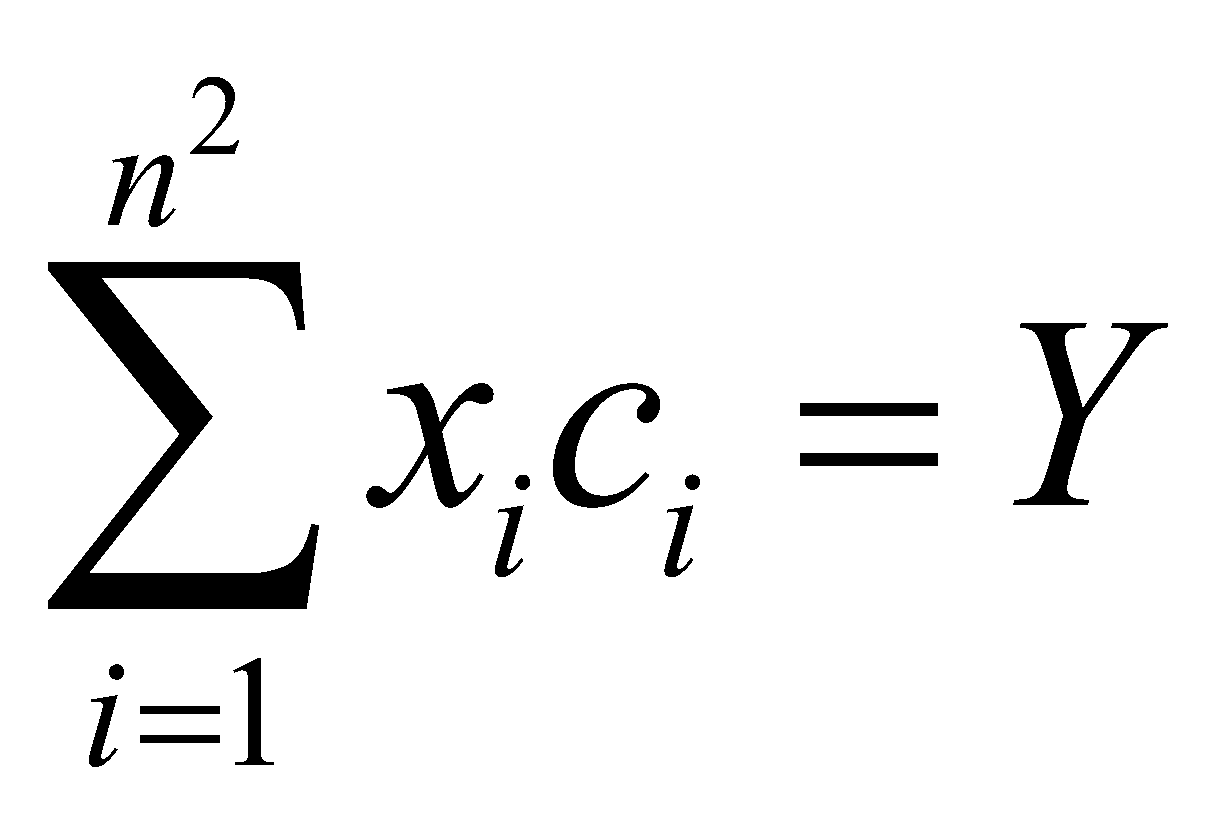
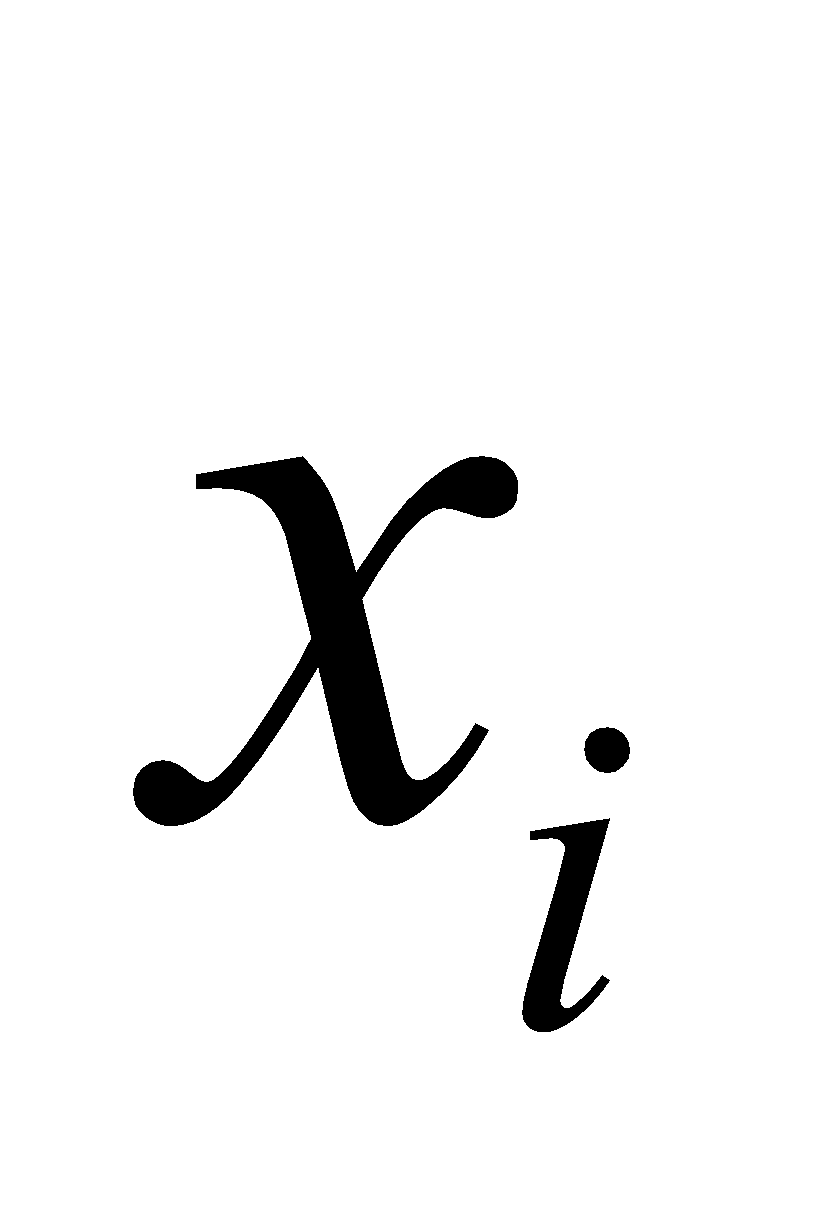
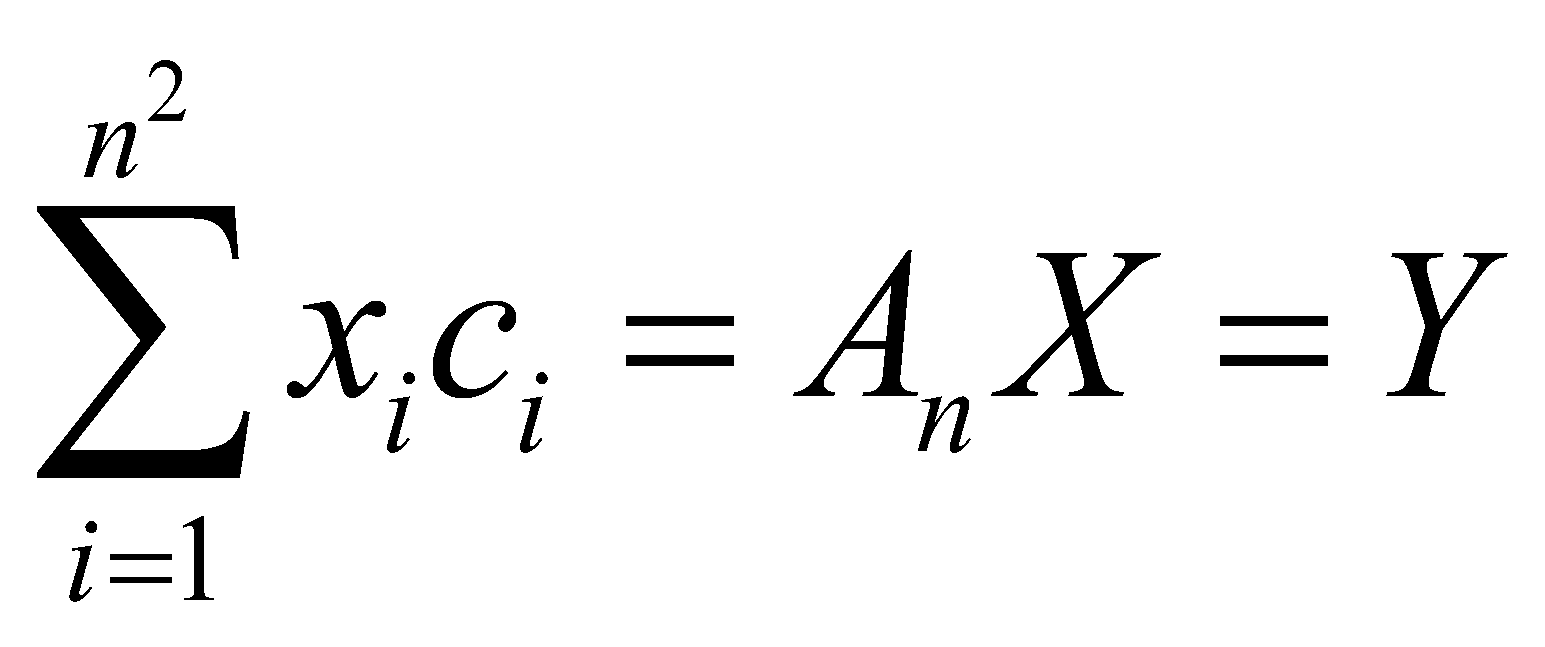
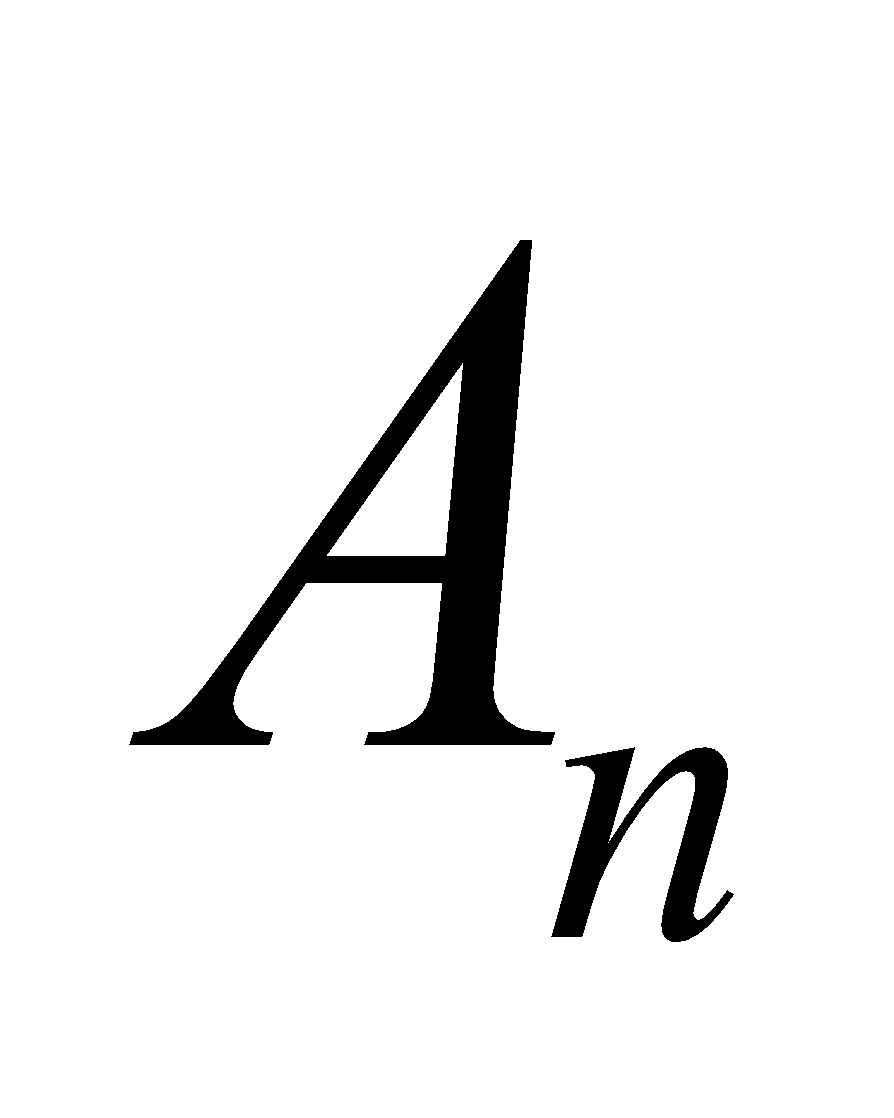
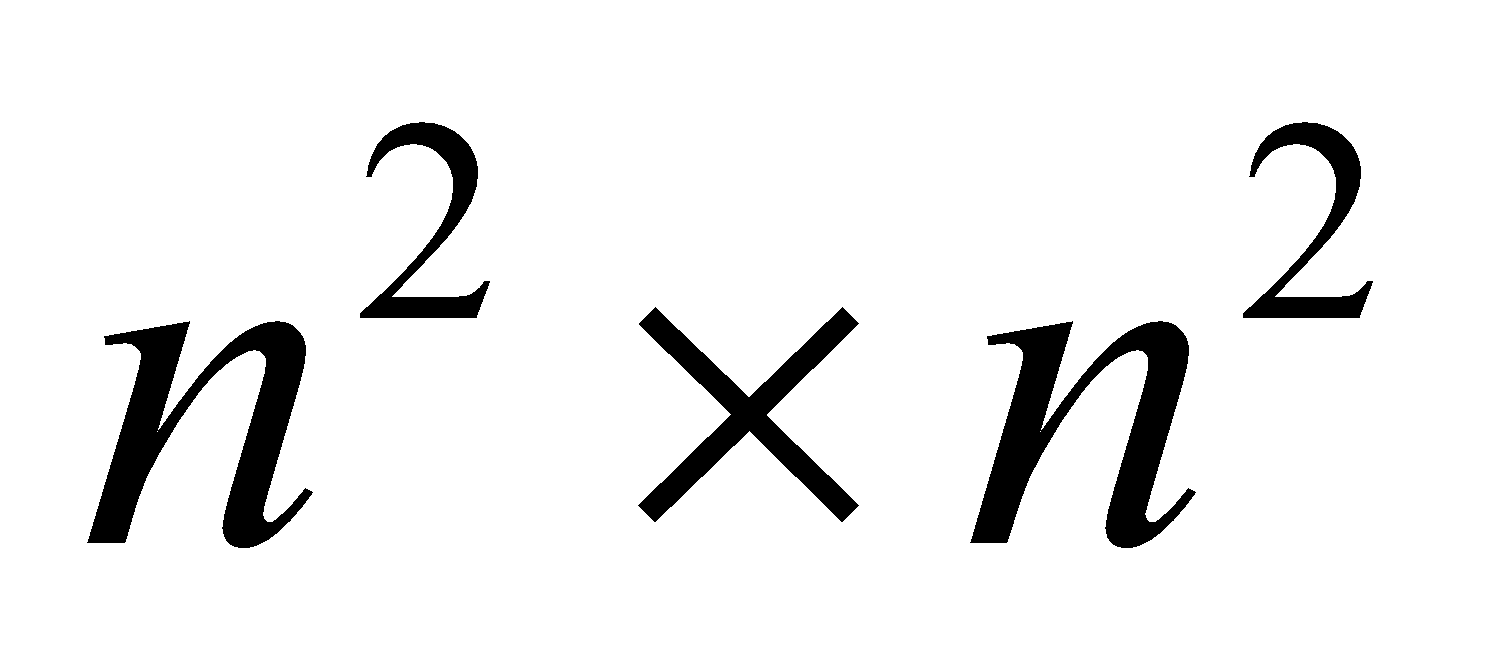
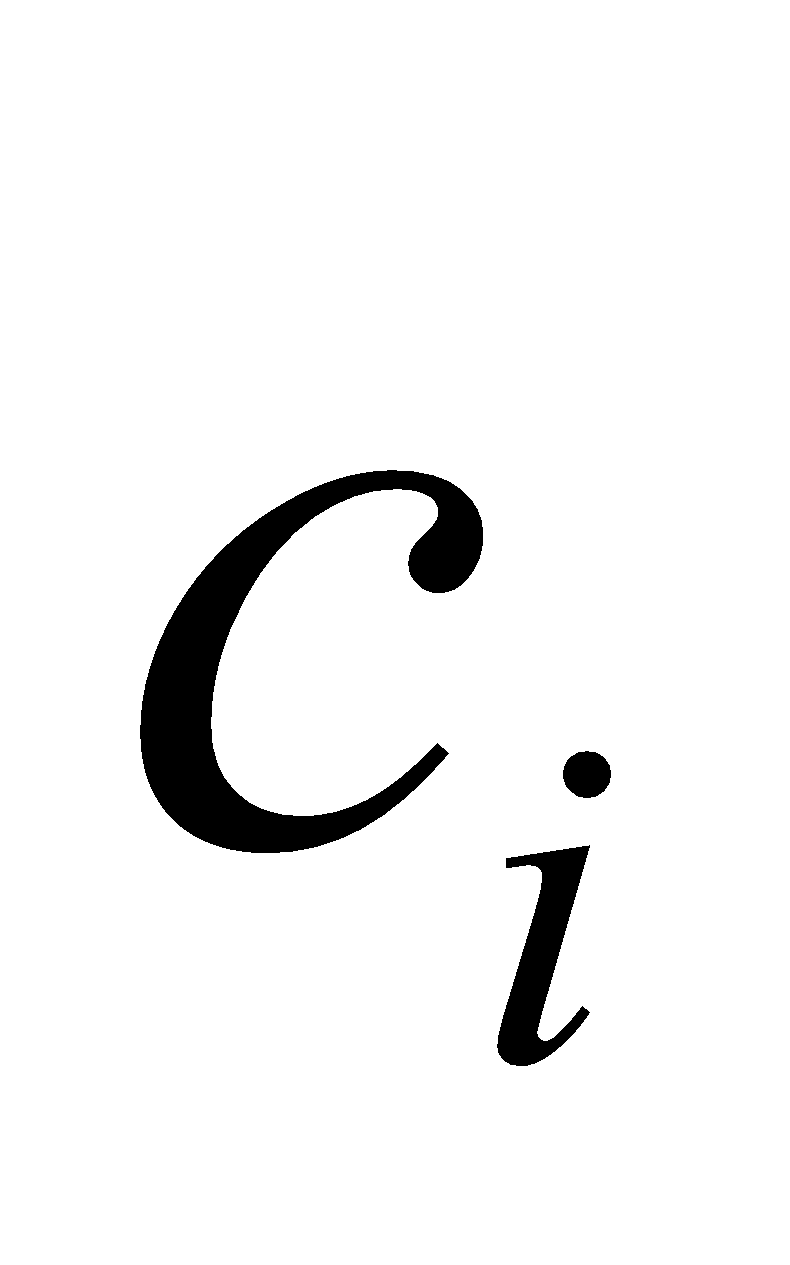
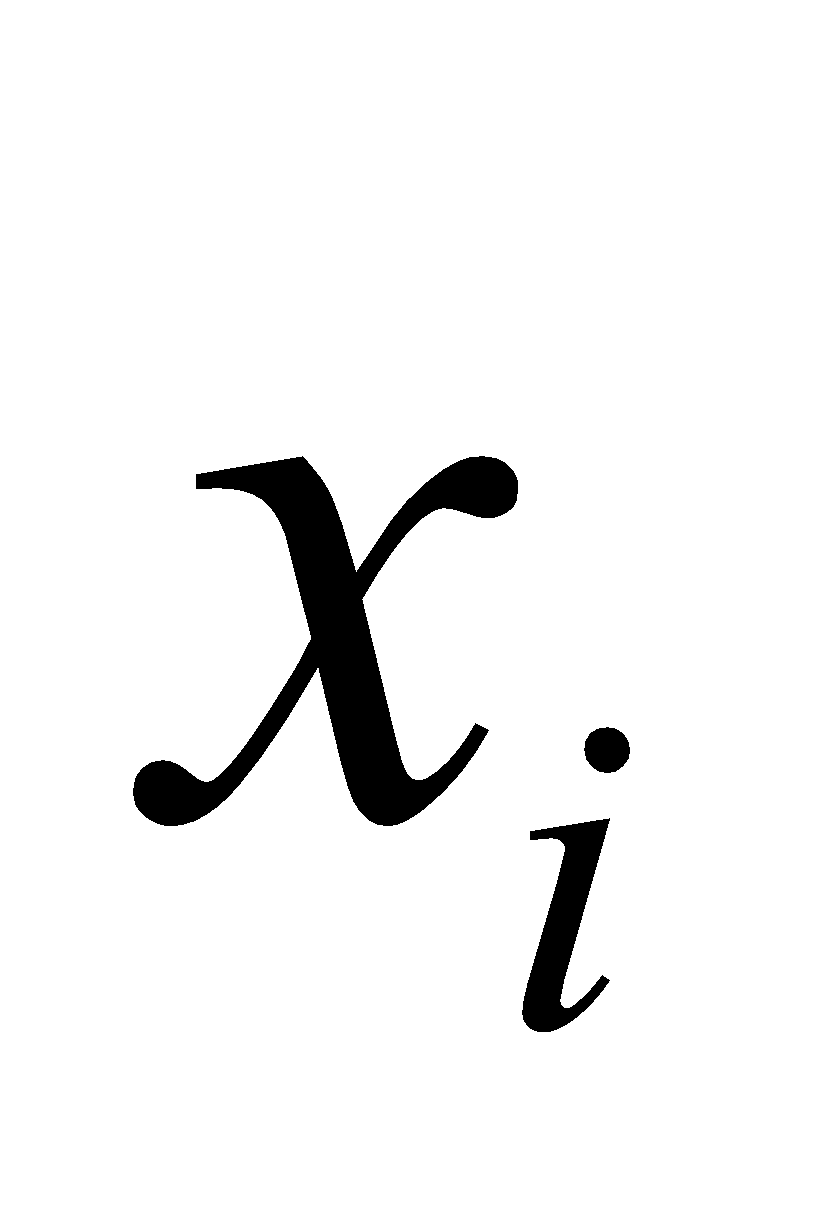
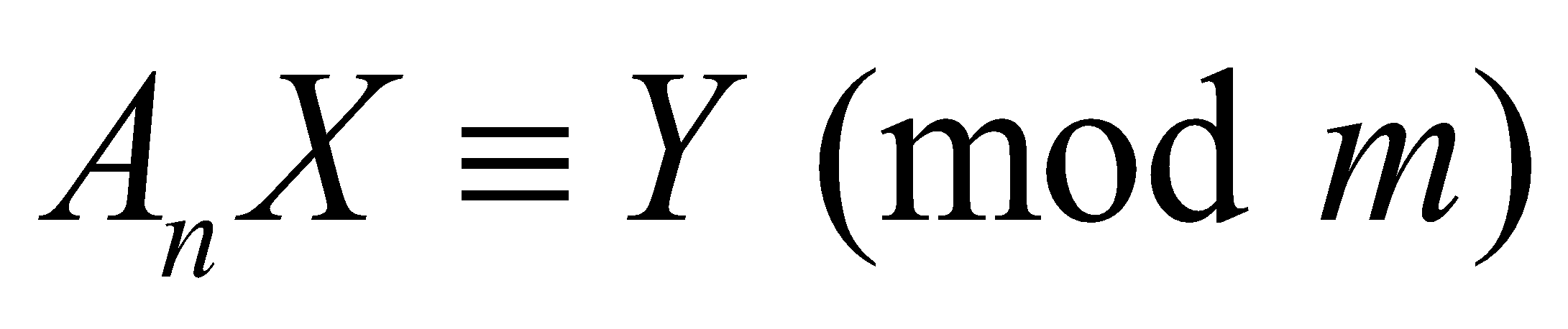
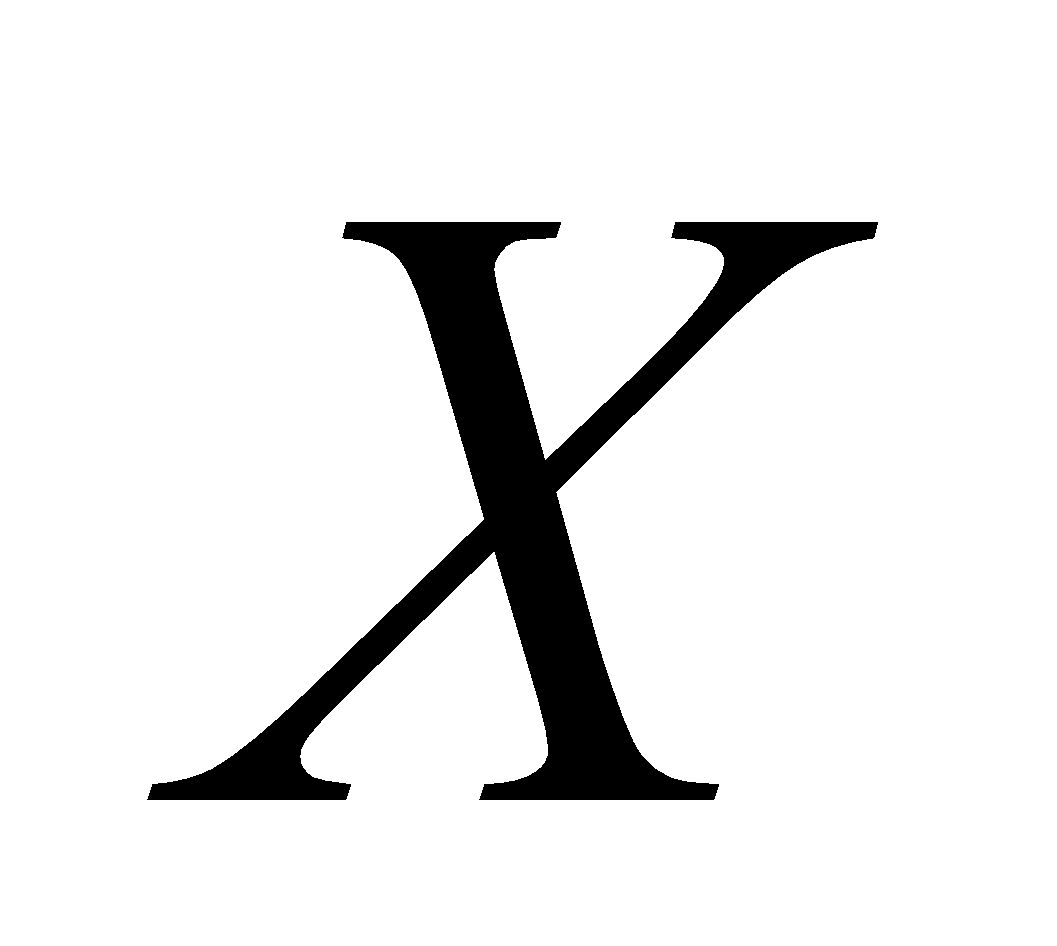
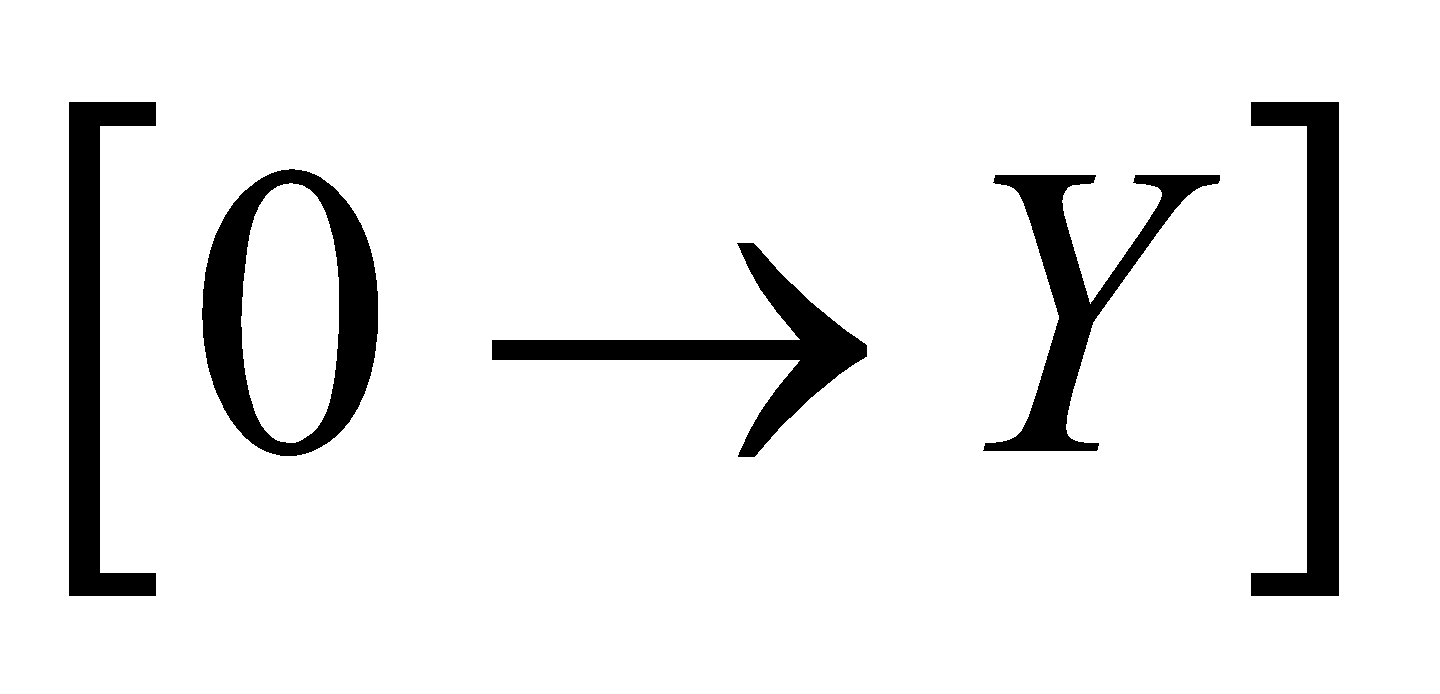
Раттанакарм С., Йоосири Б., Пудки Дж., Аморнватчарапонг А.

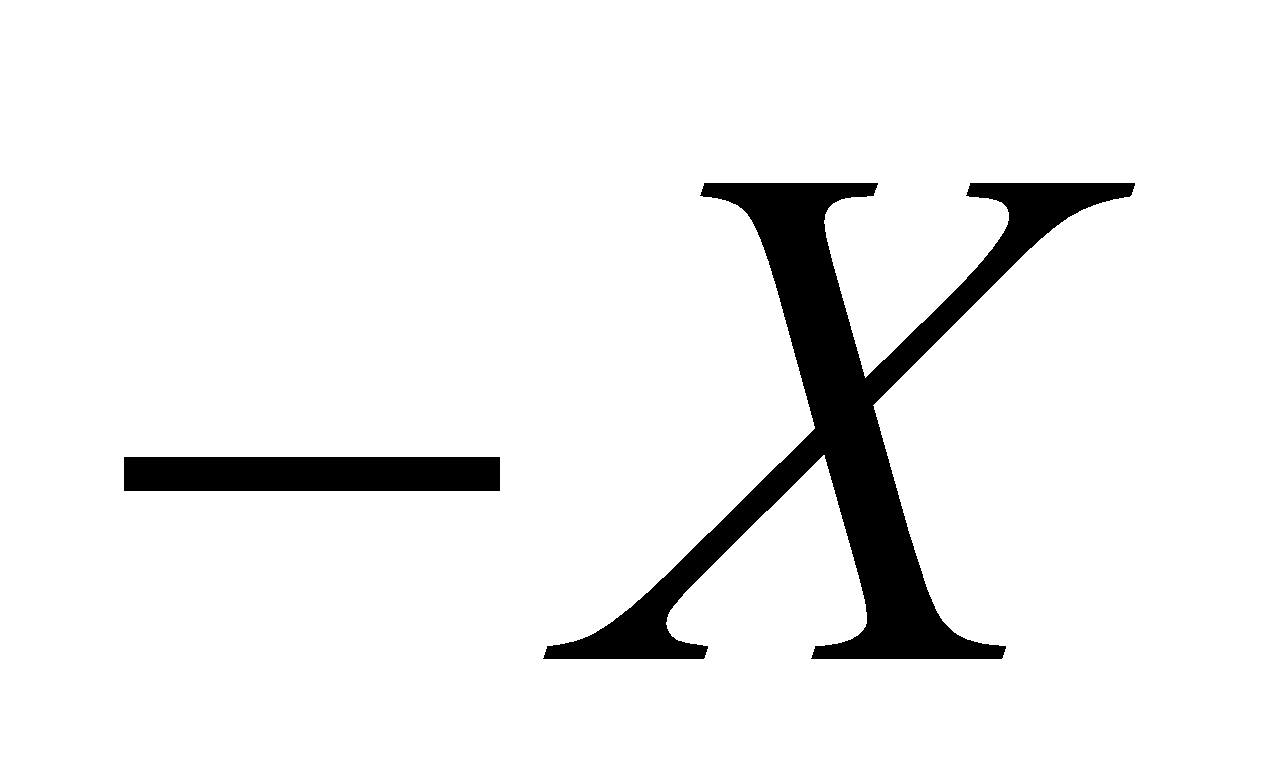
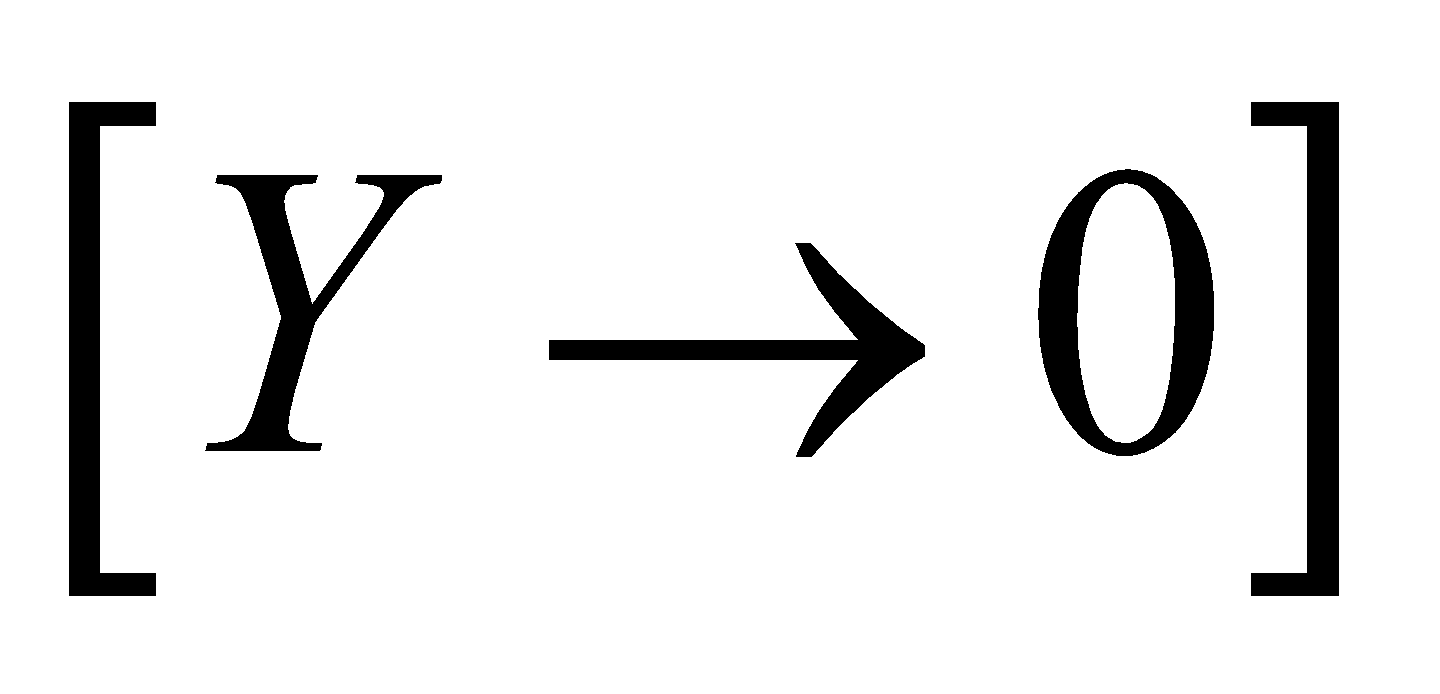
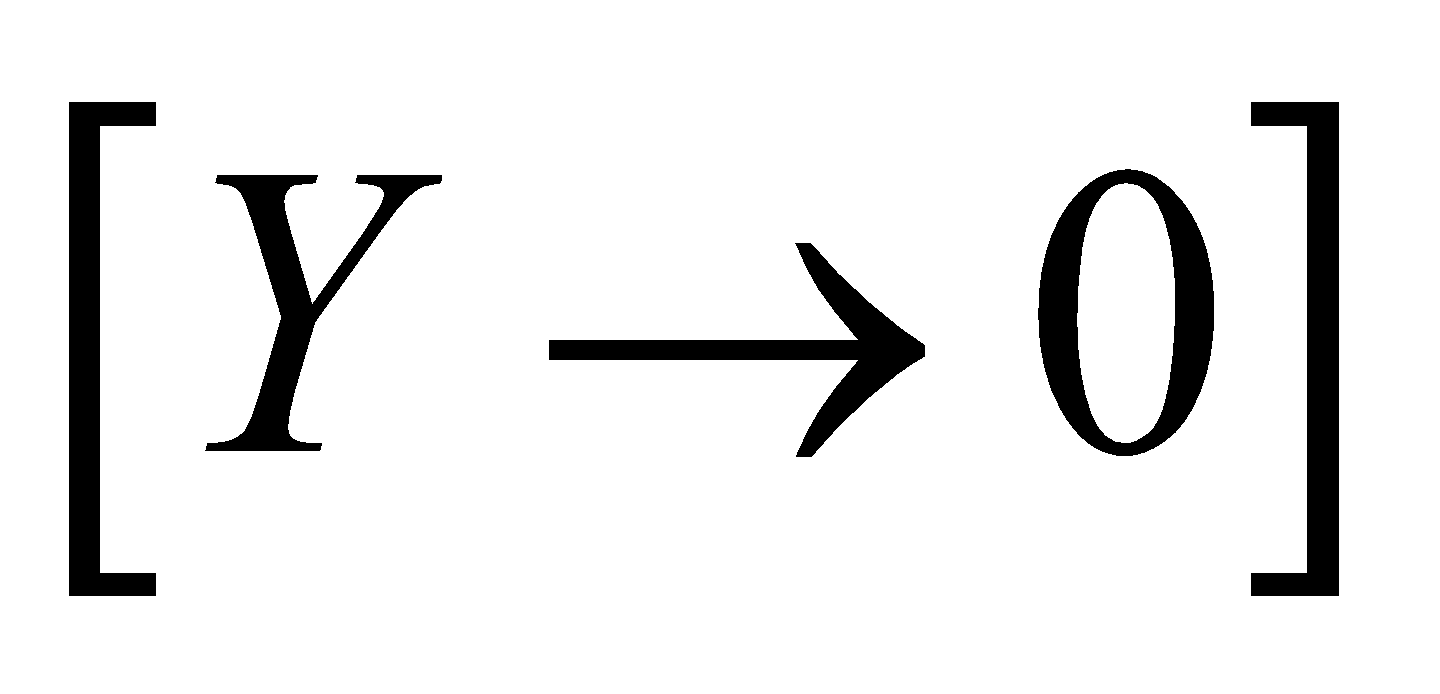
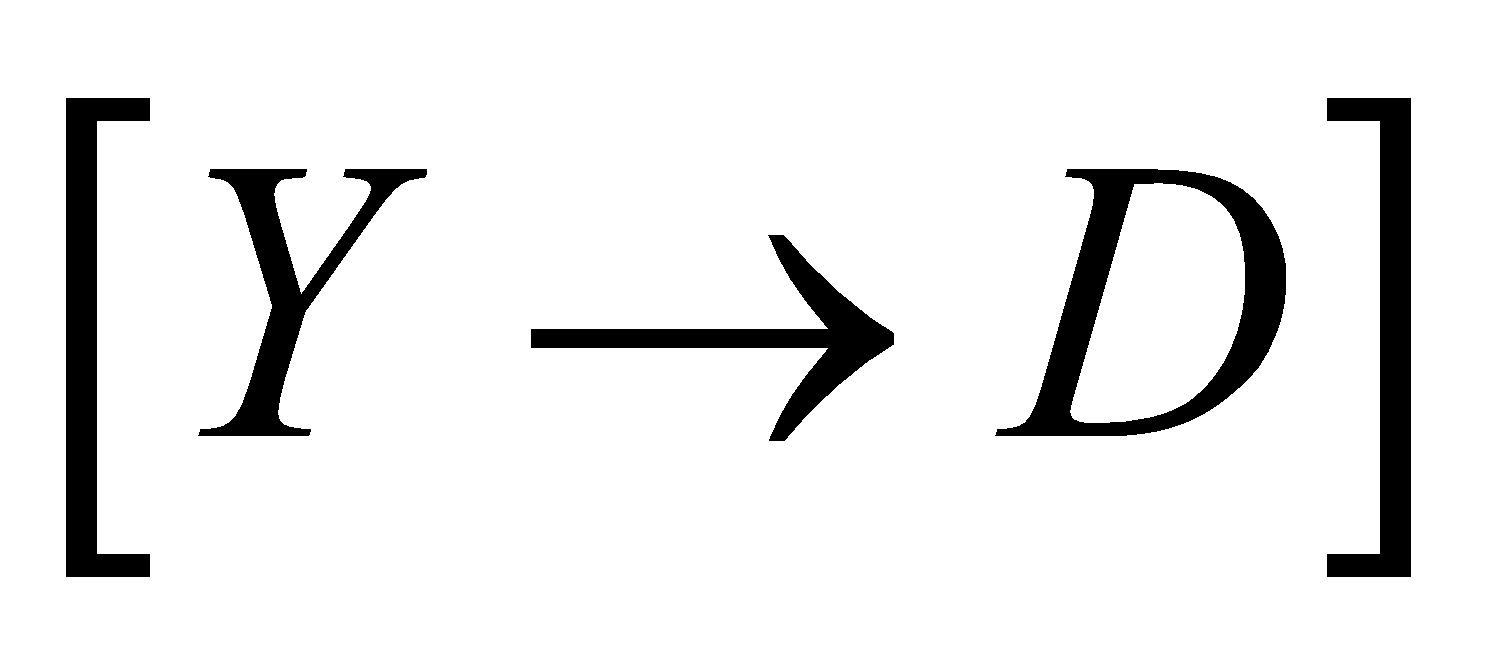
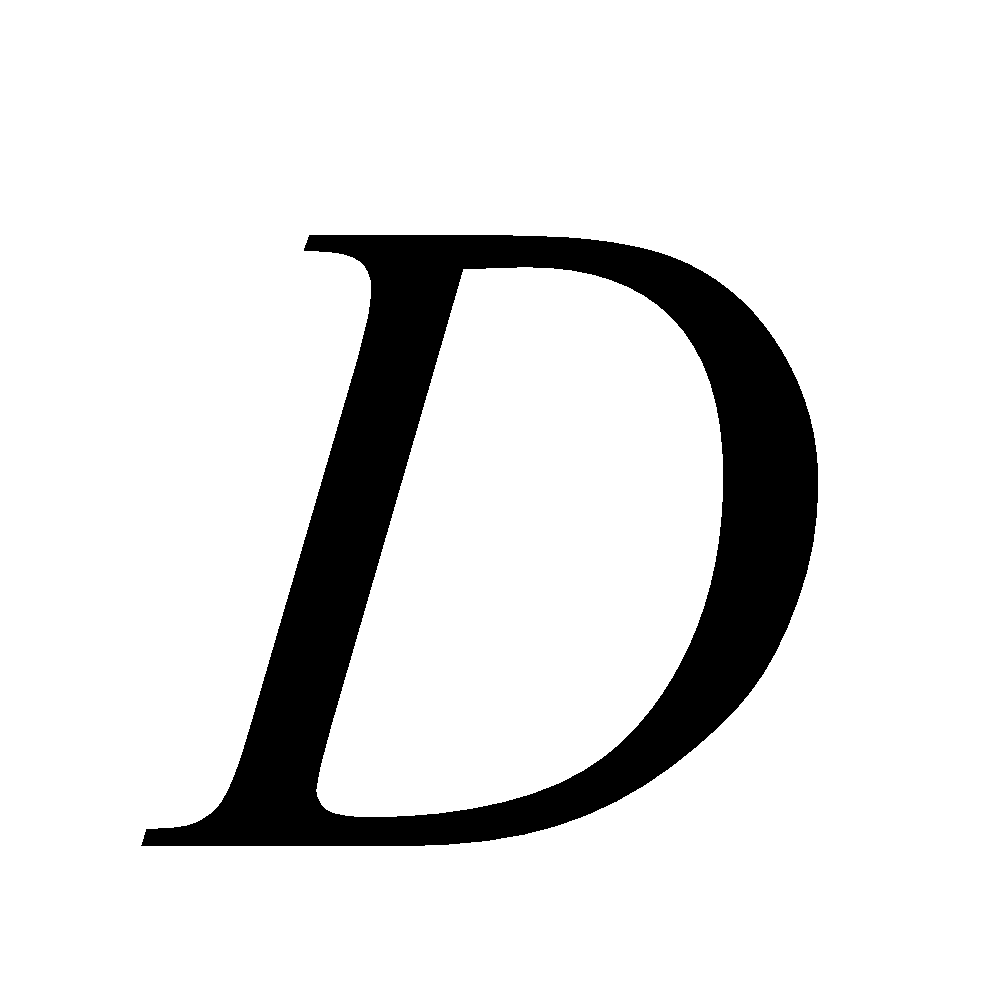
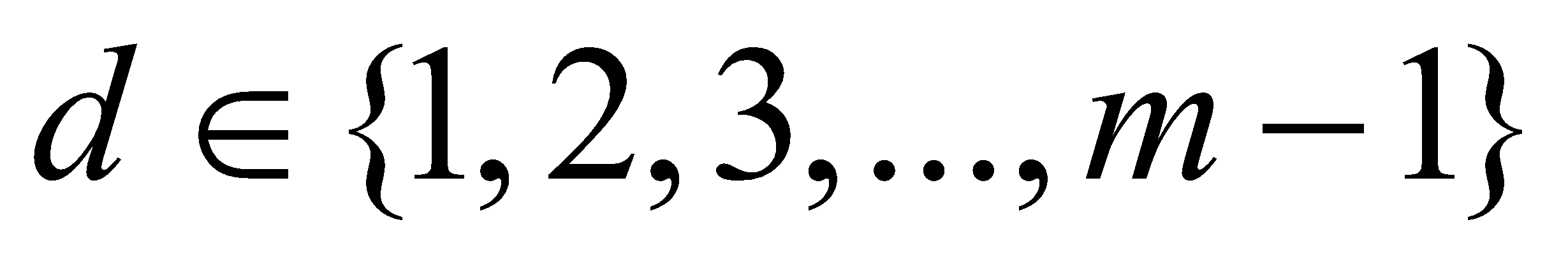
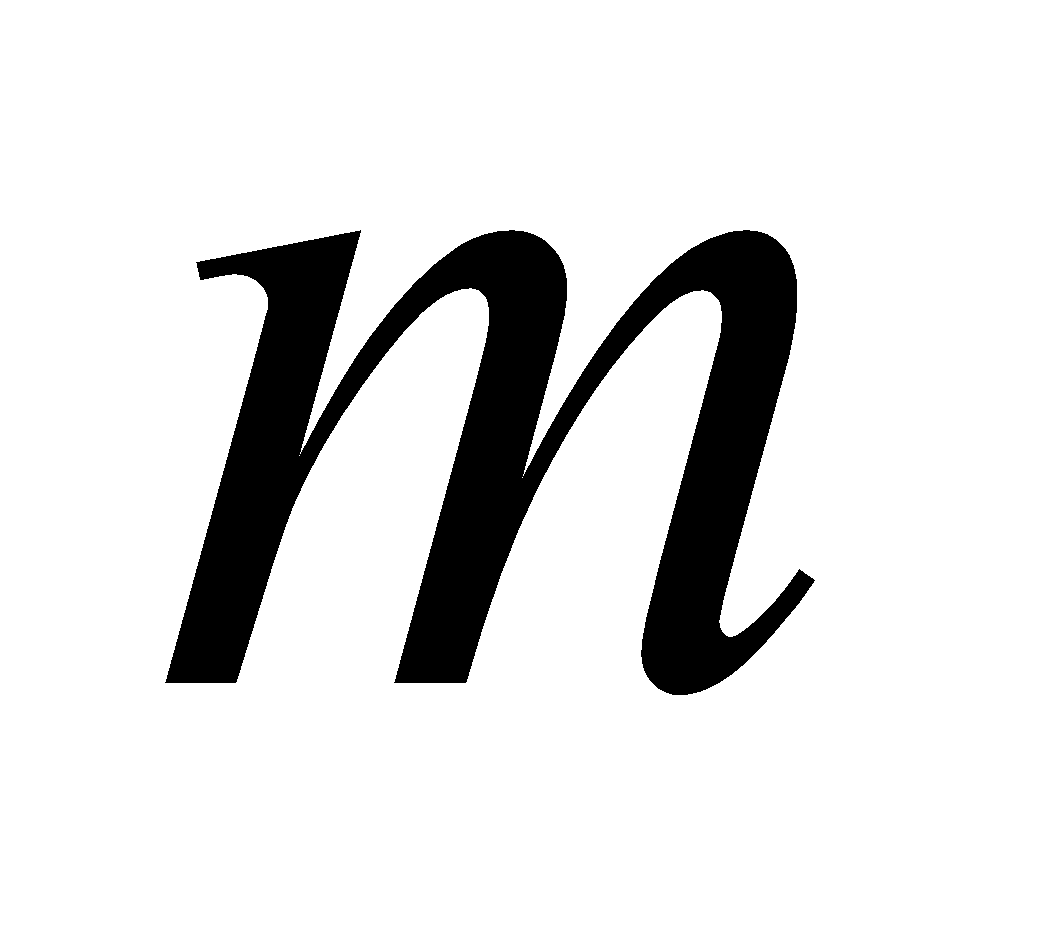
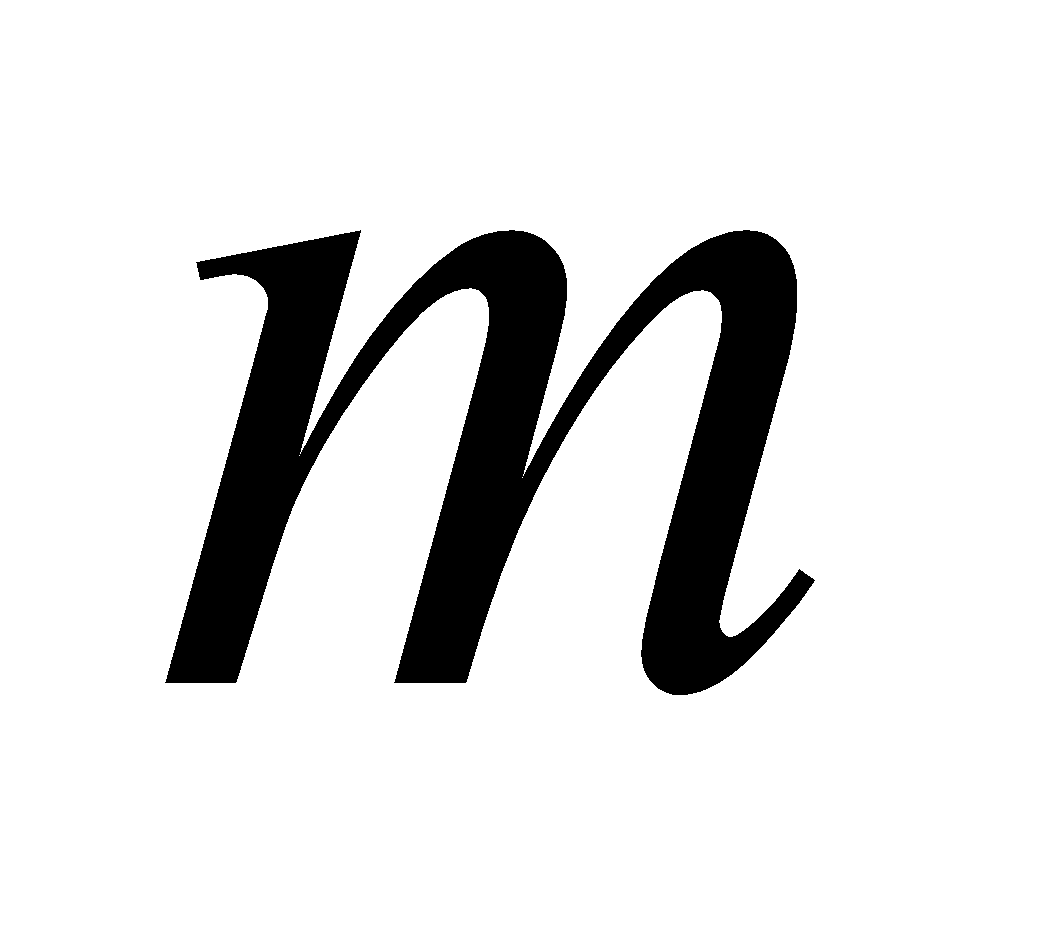
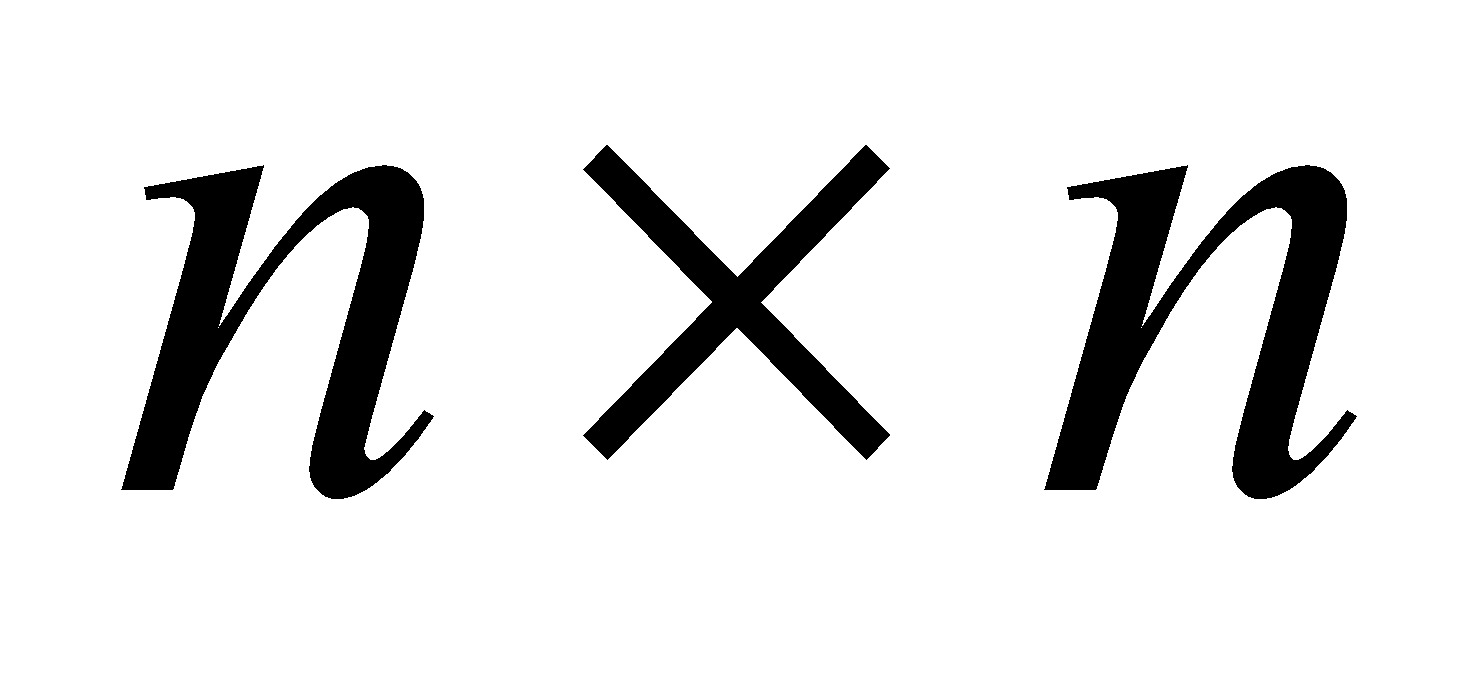
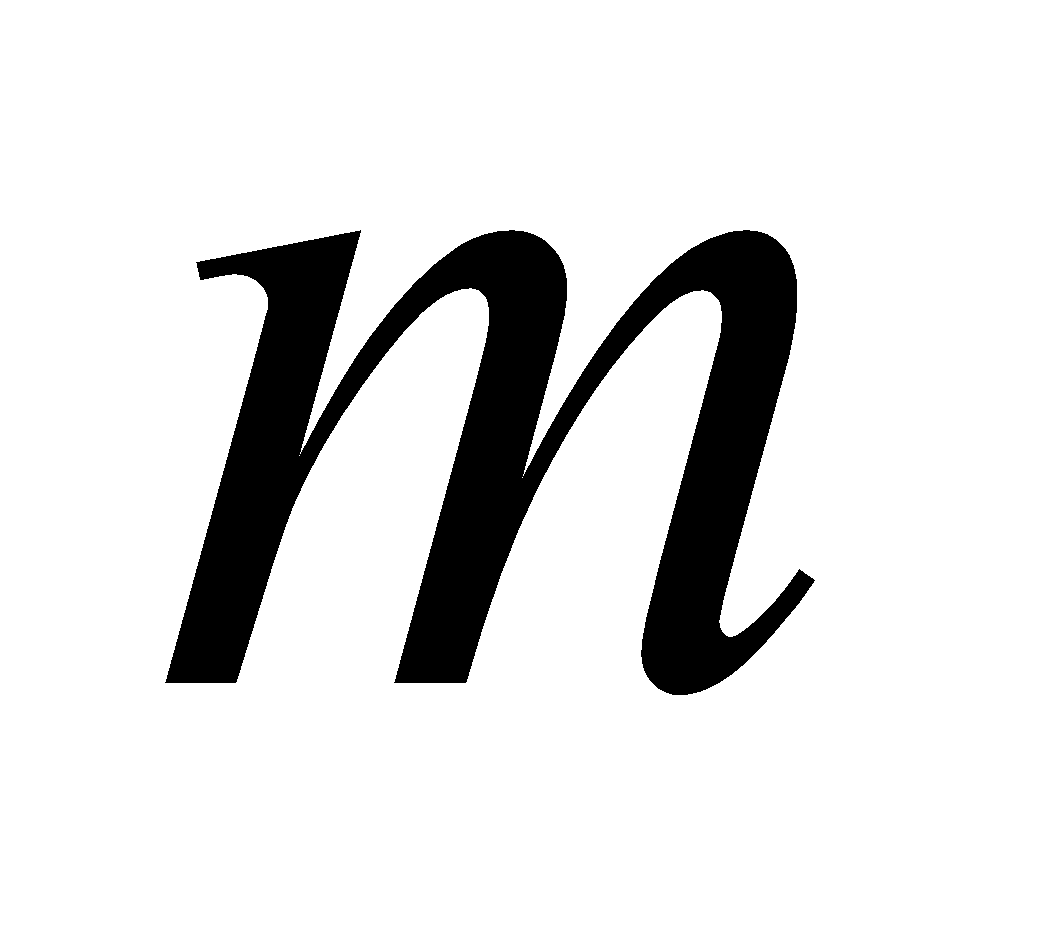
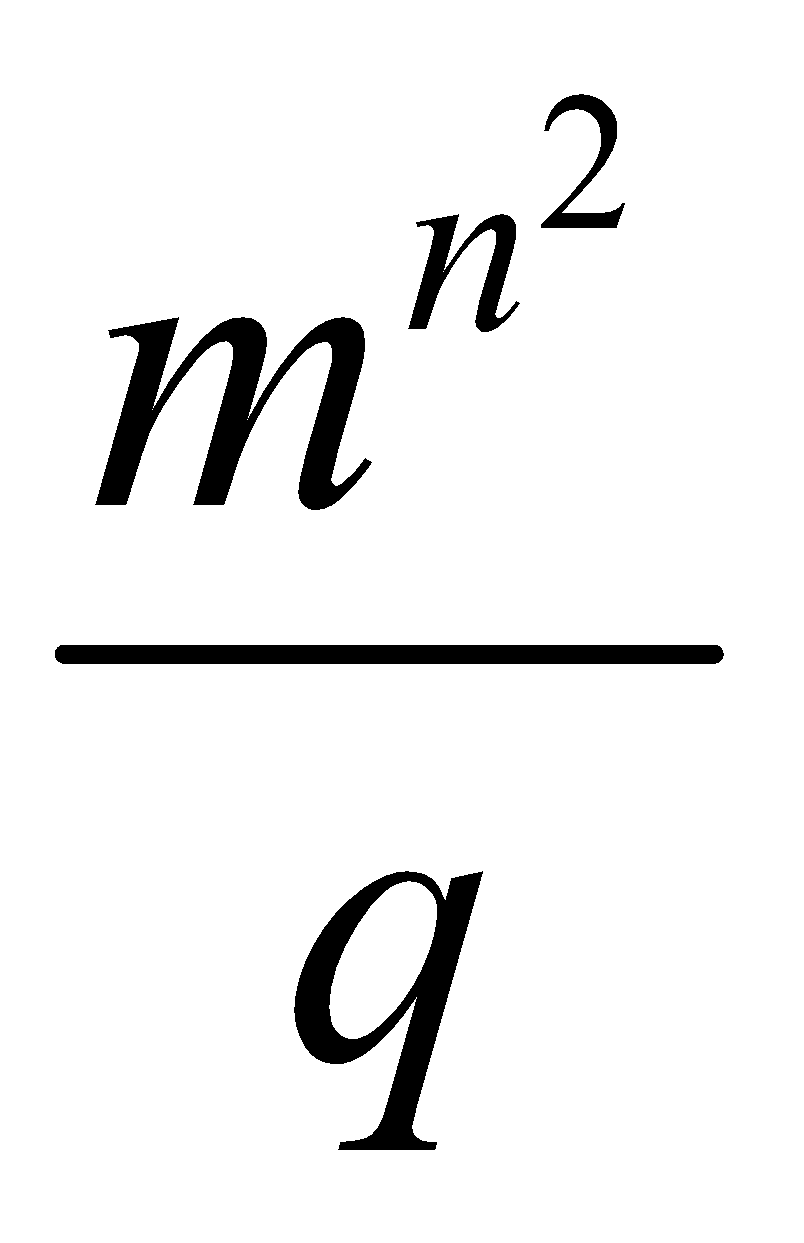
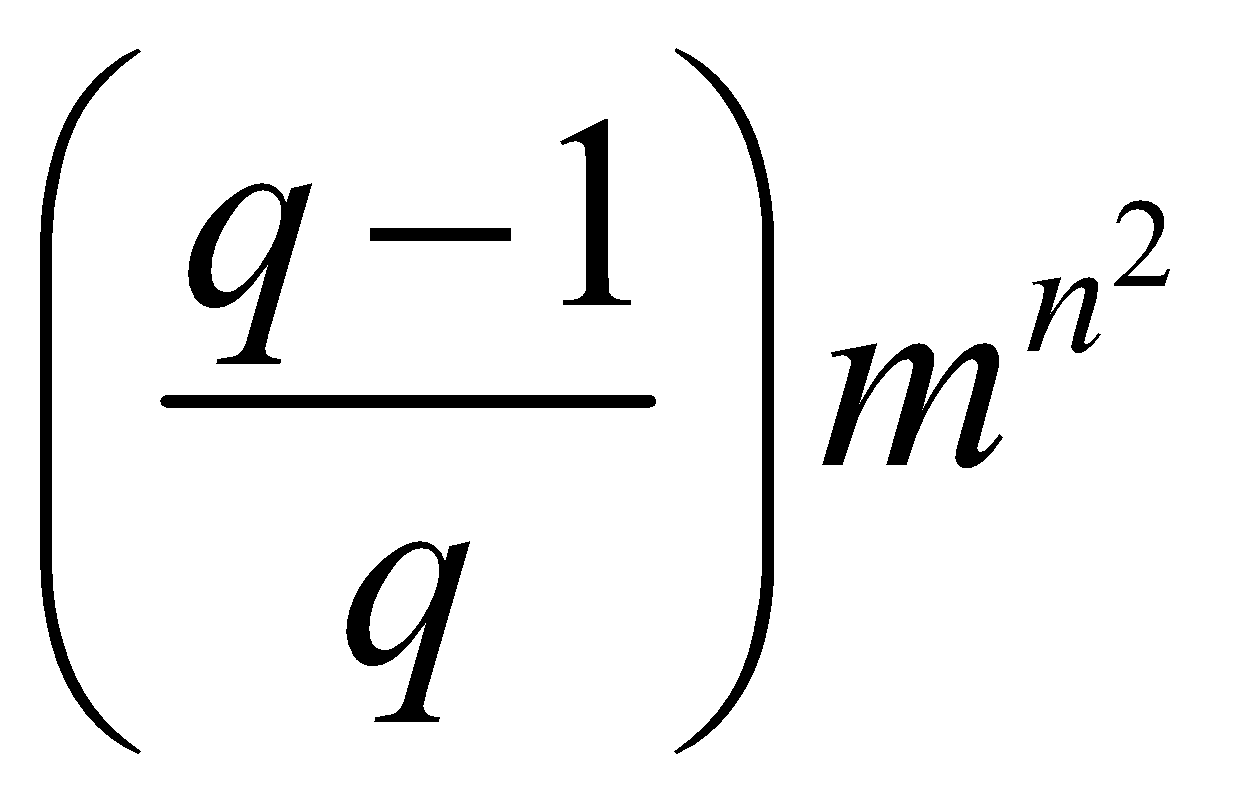
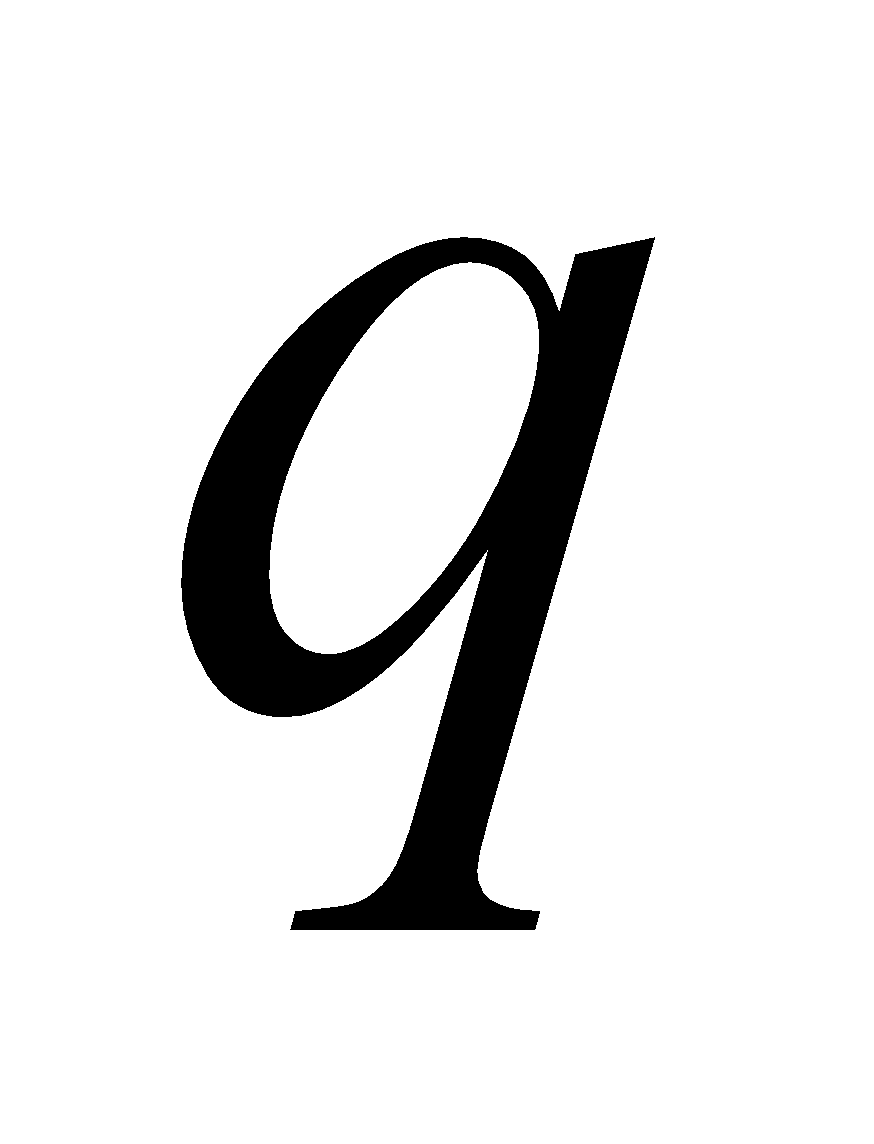
*Школа Махидол Виттаянусорн, Након Патом, Таиланд*

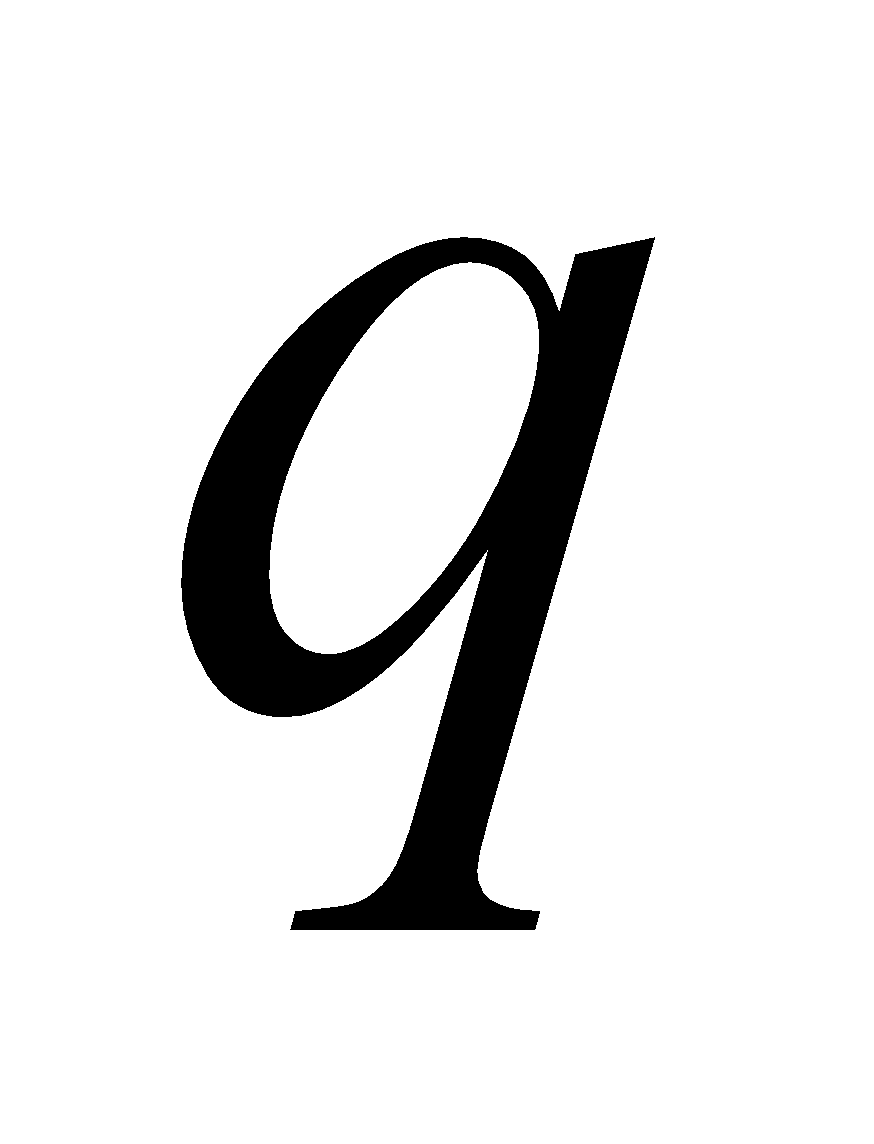
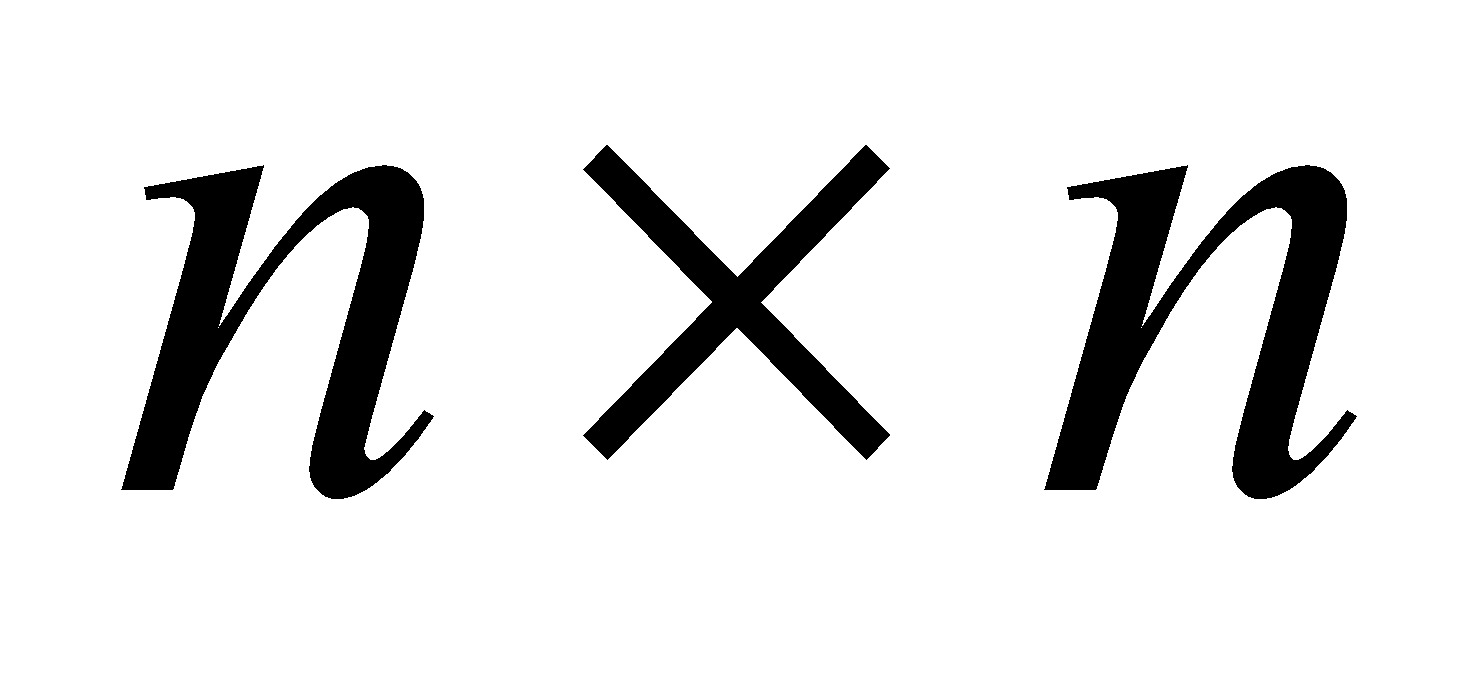
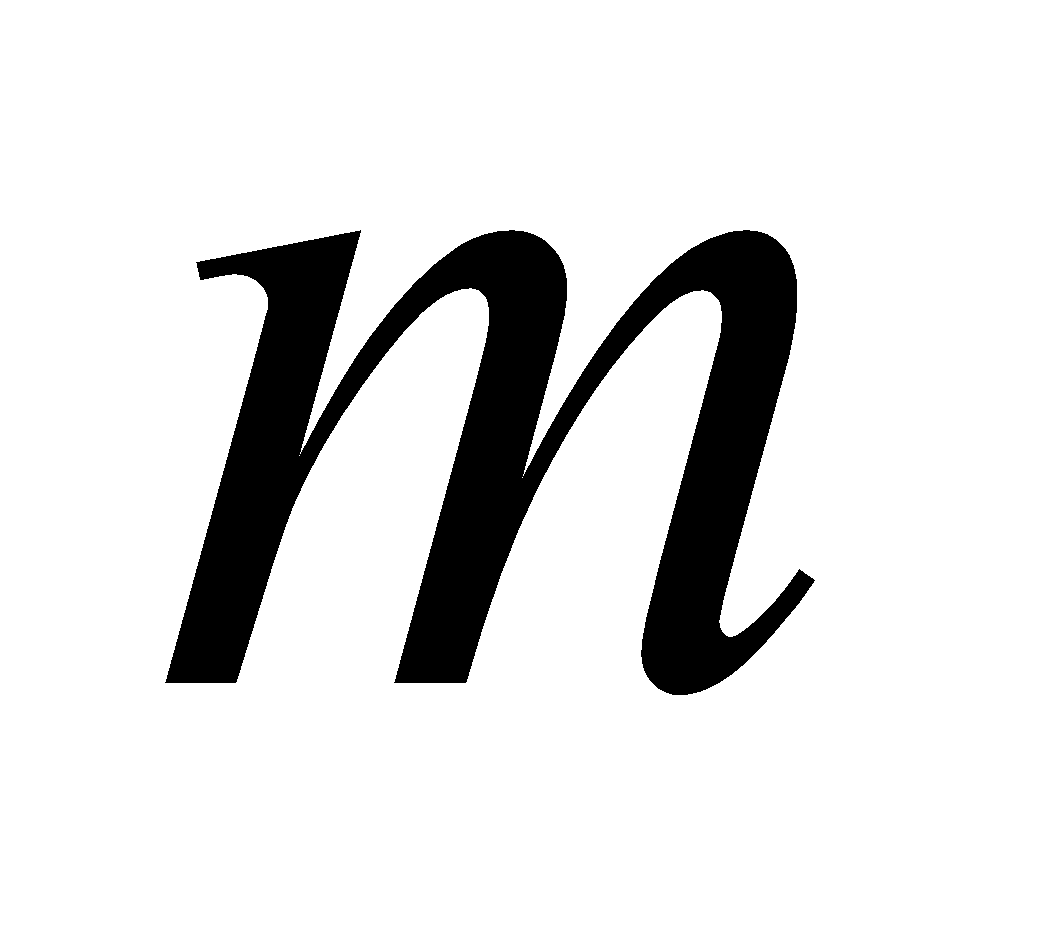
*sitchai0645@gmail.com*

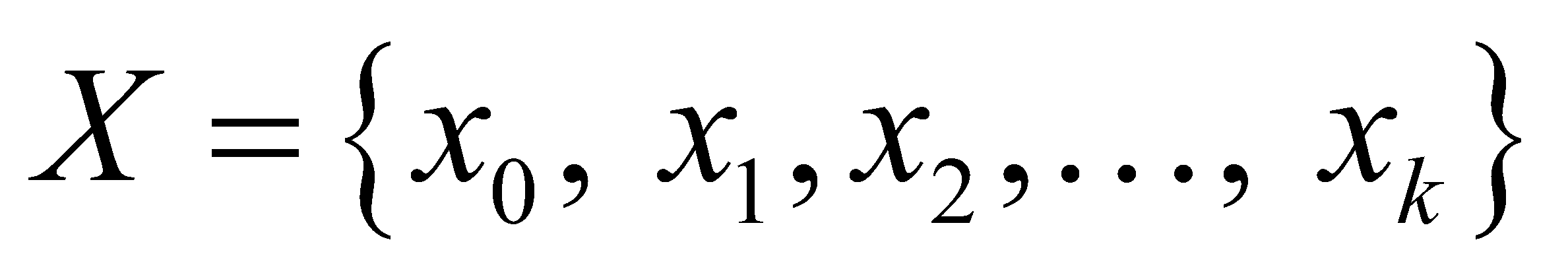
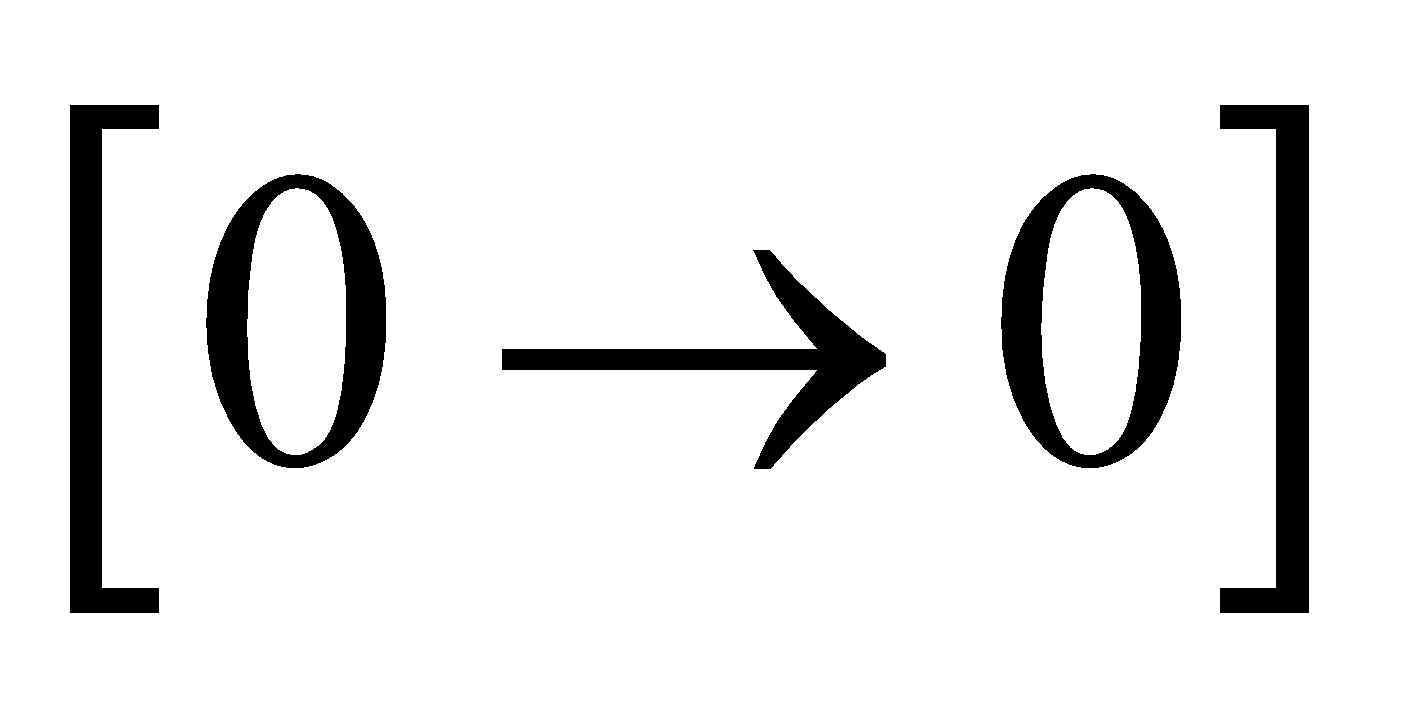
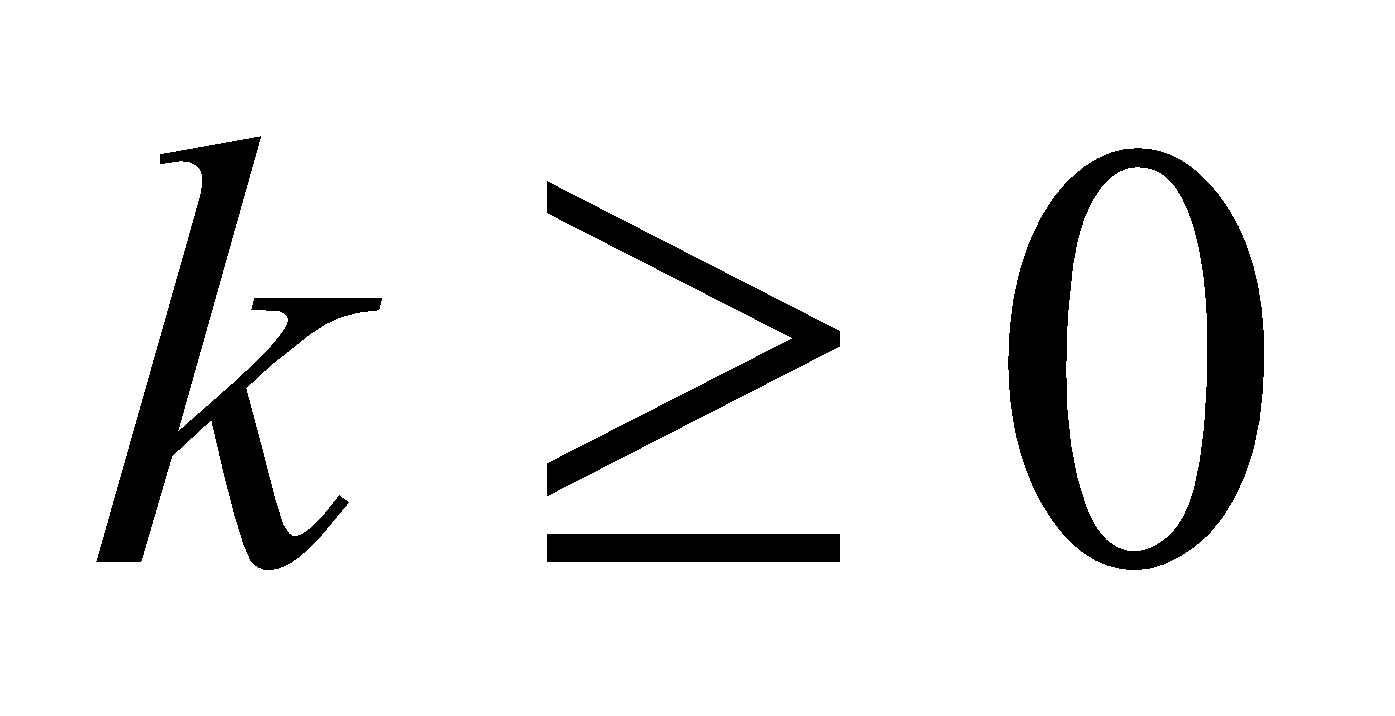
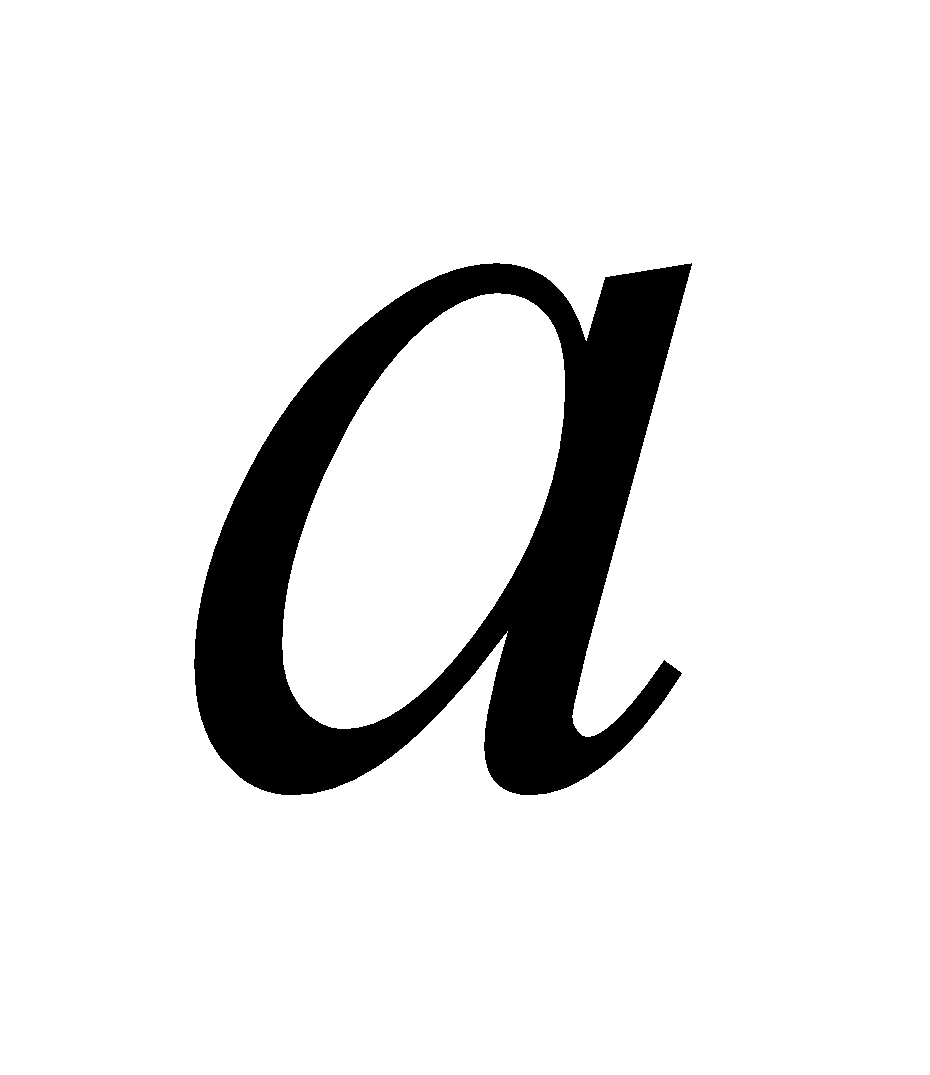
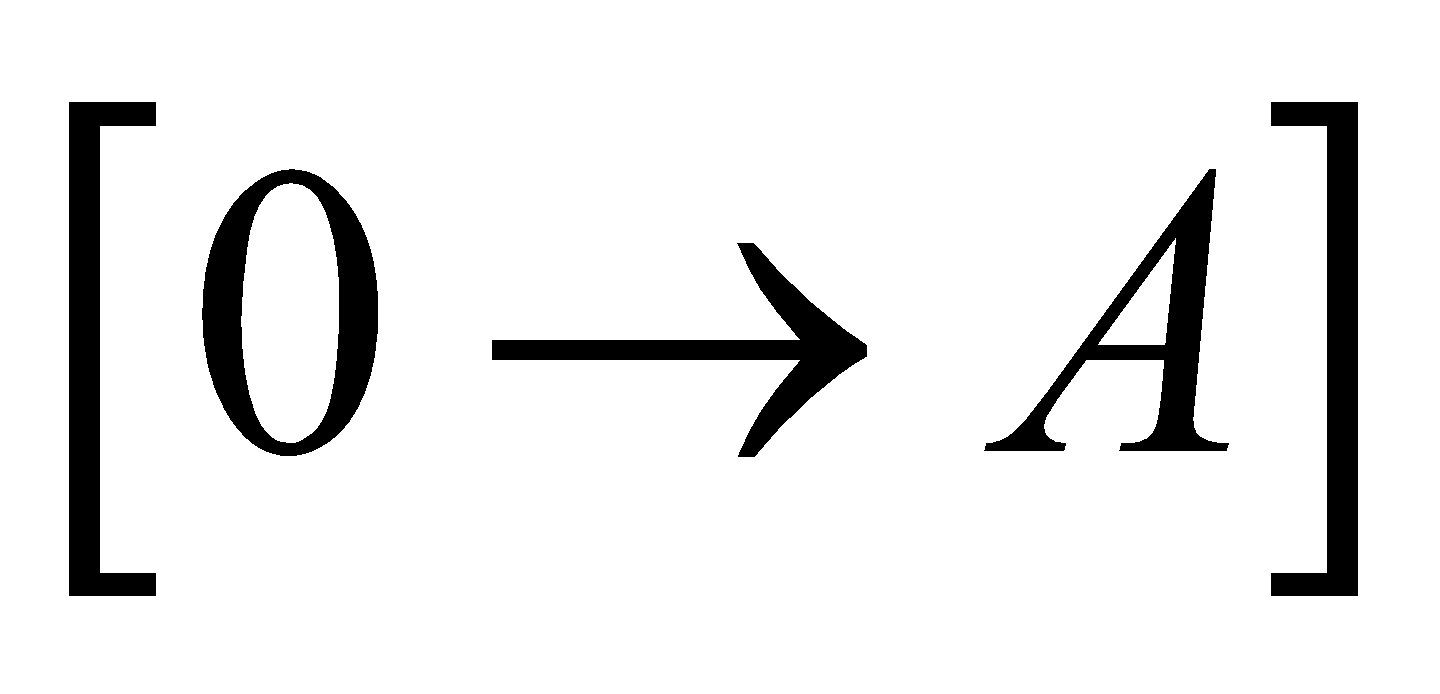
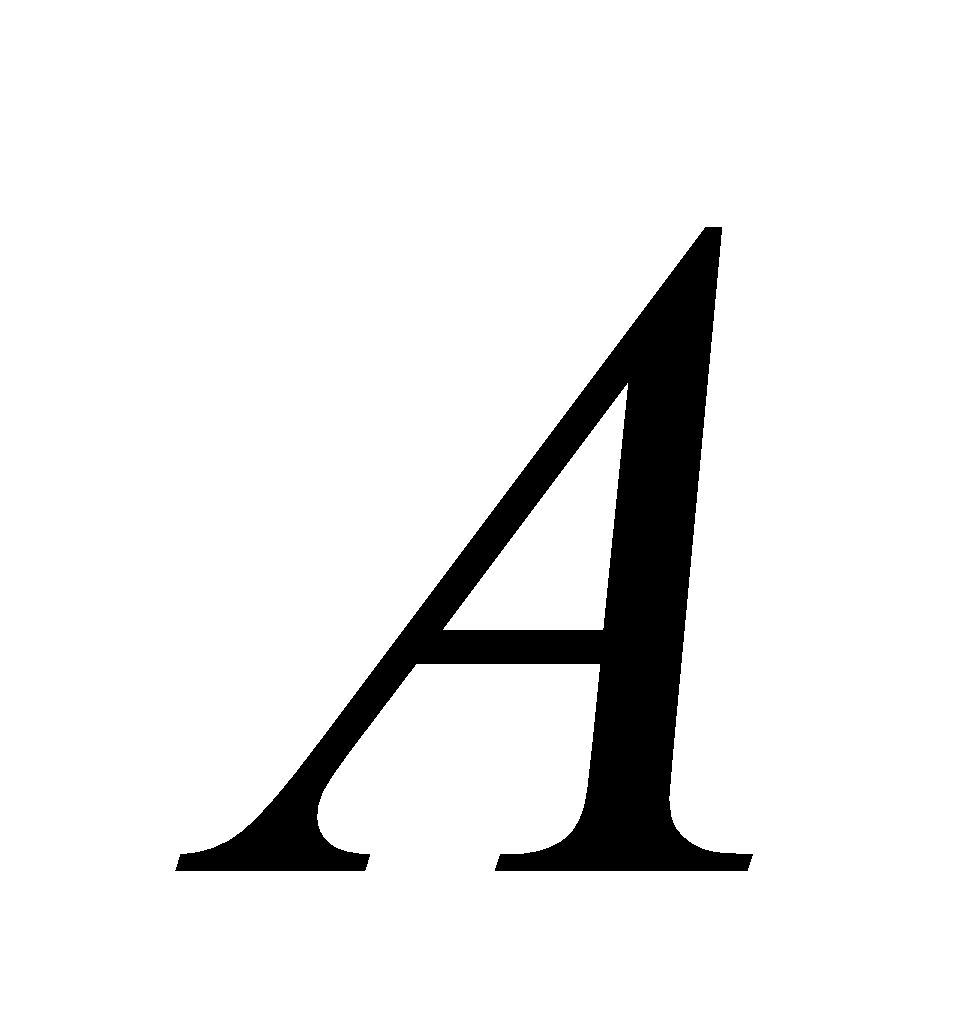
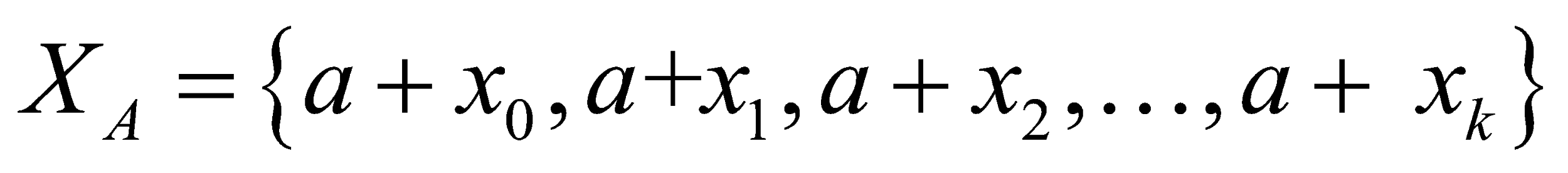
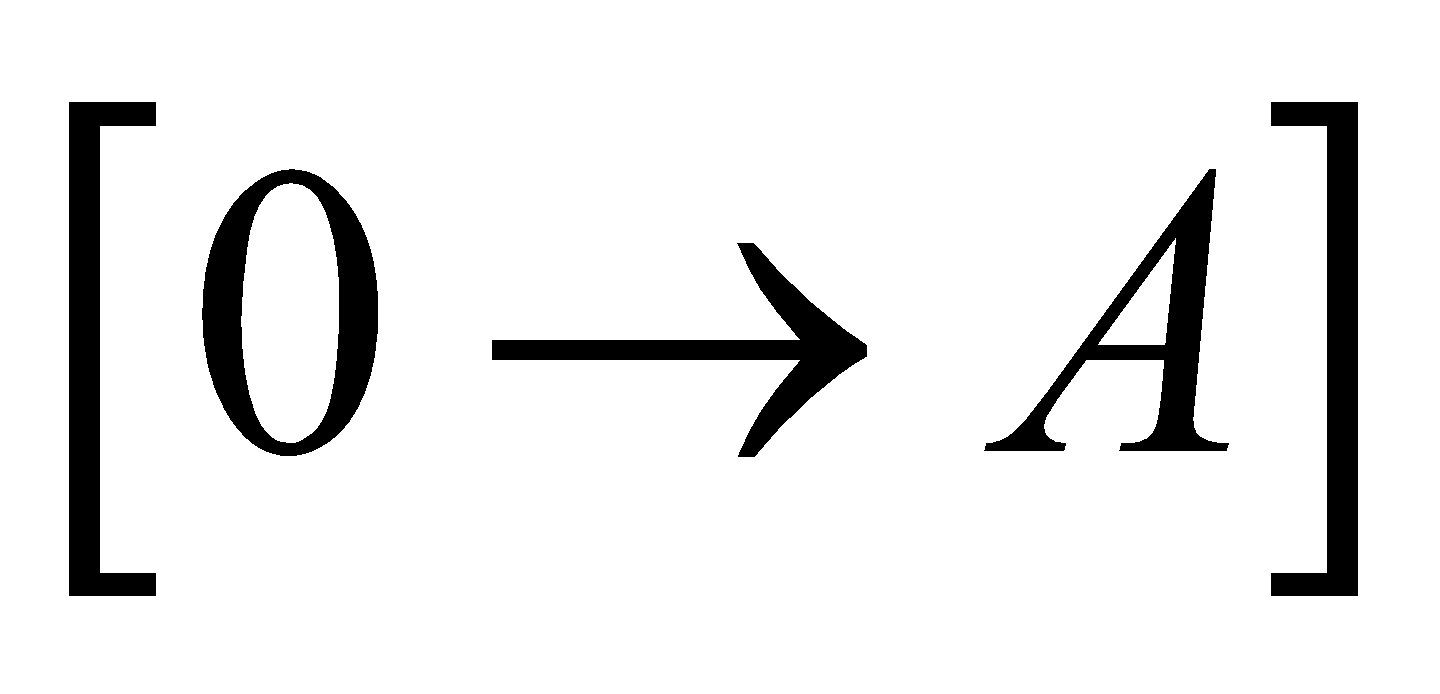
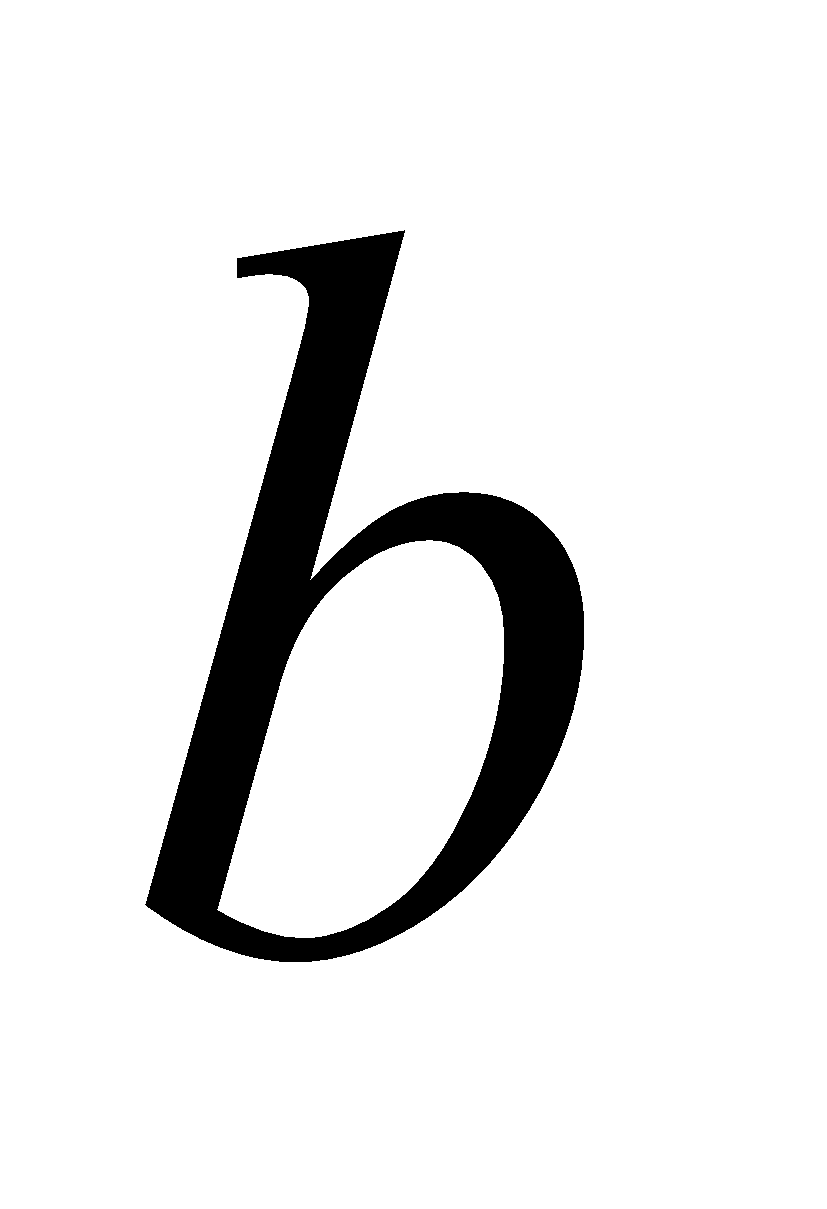
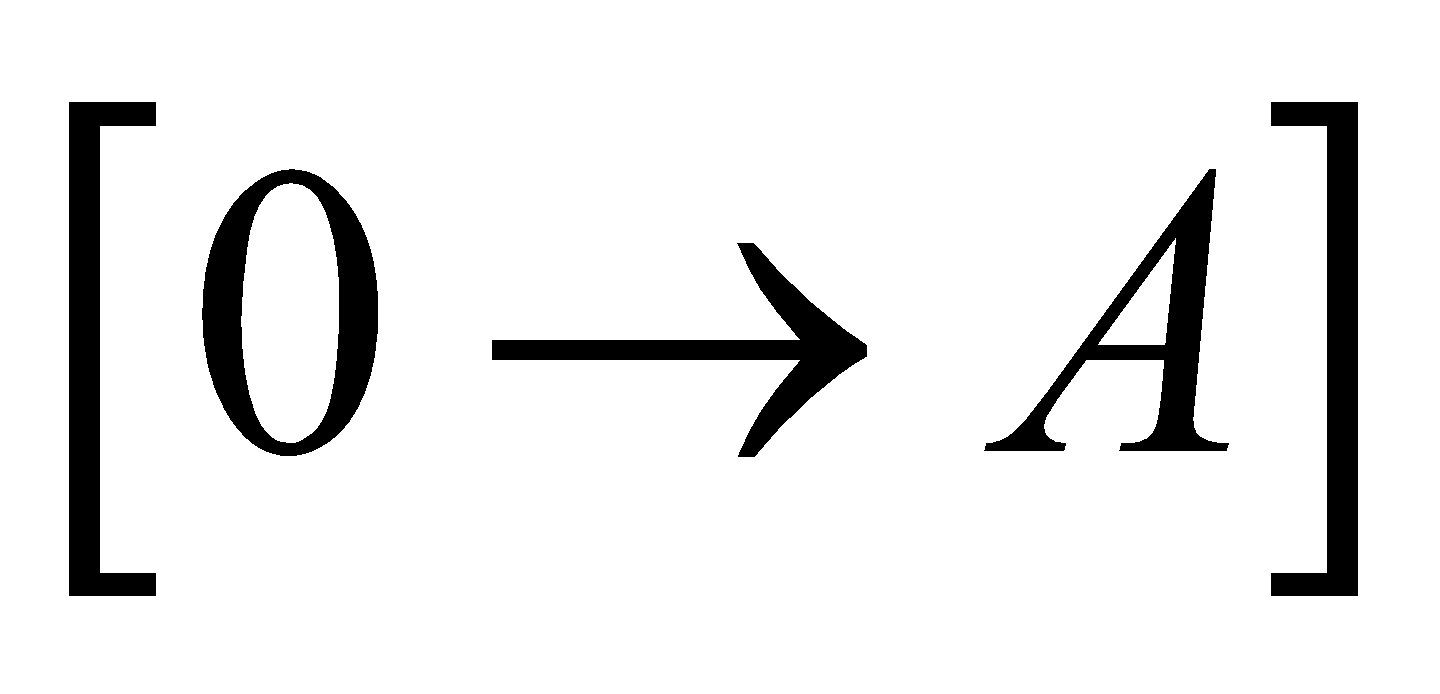
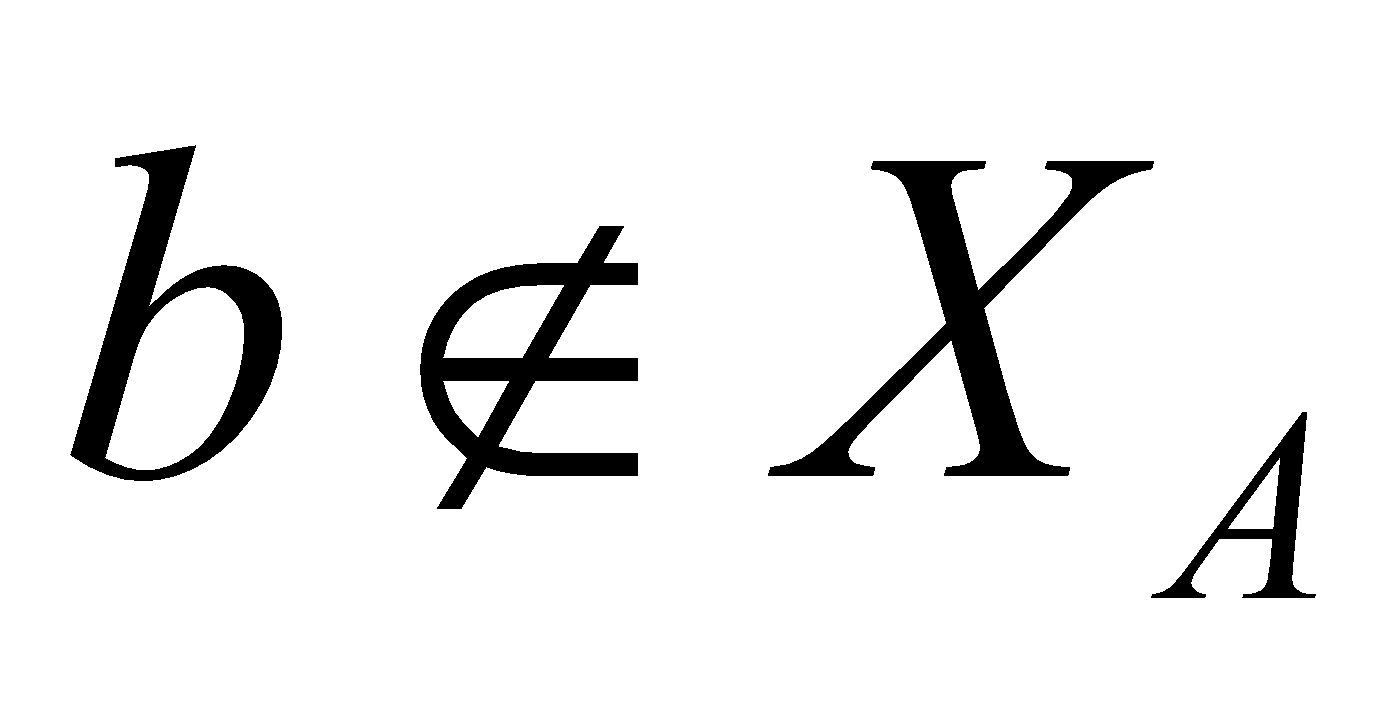
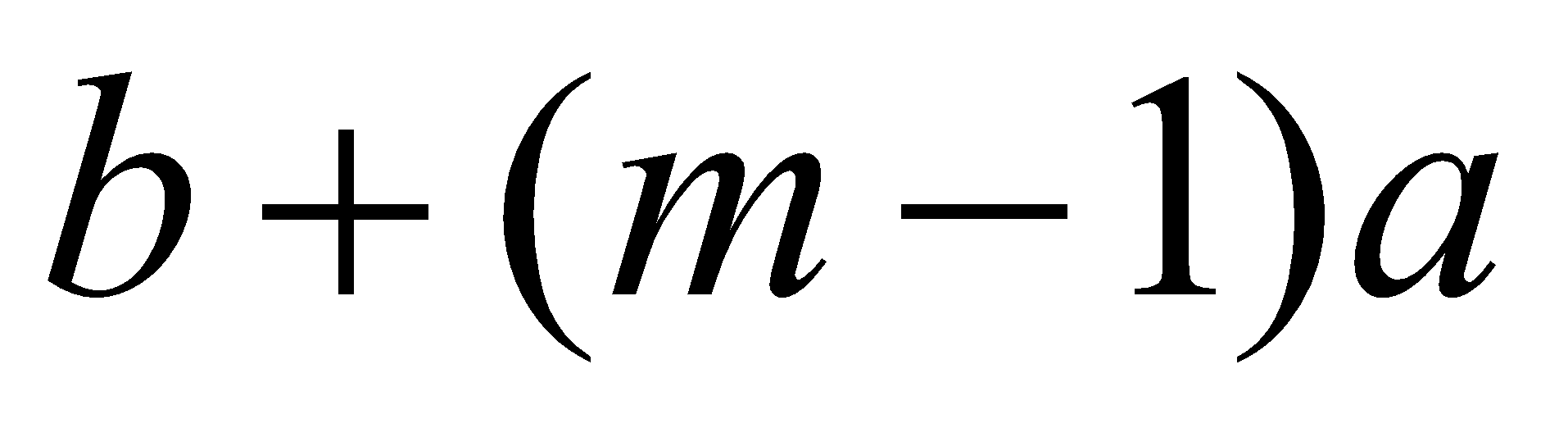
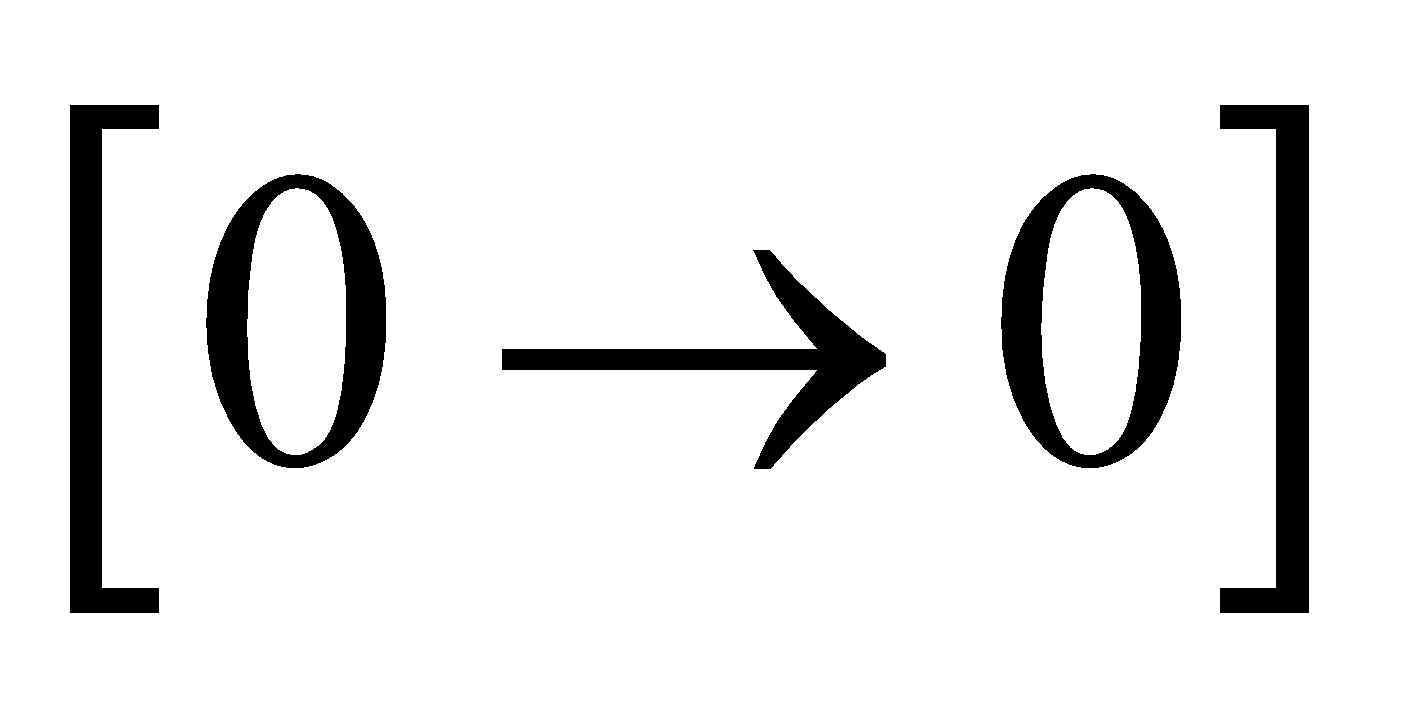
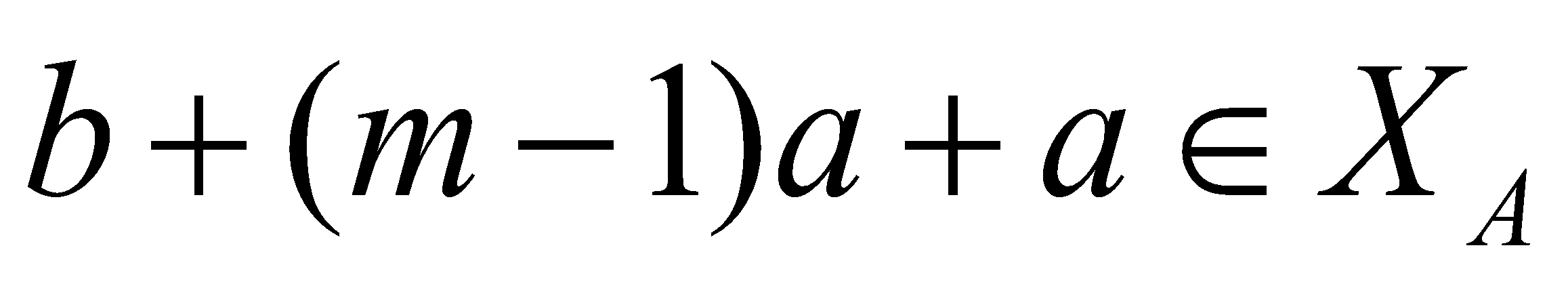
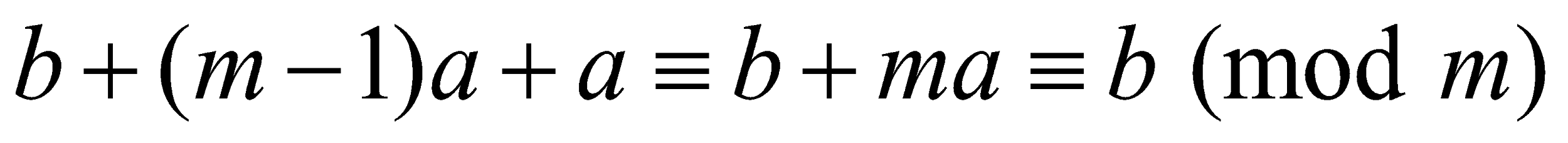
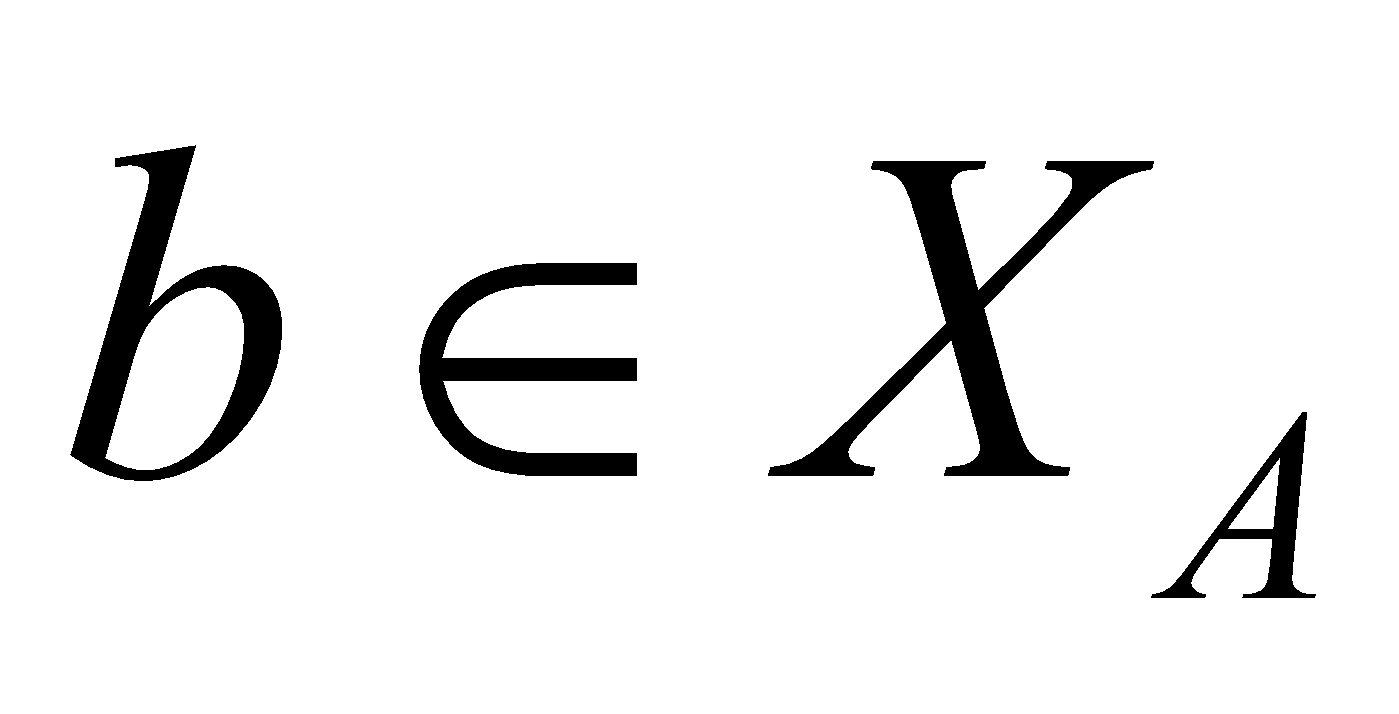
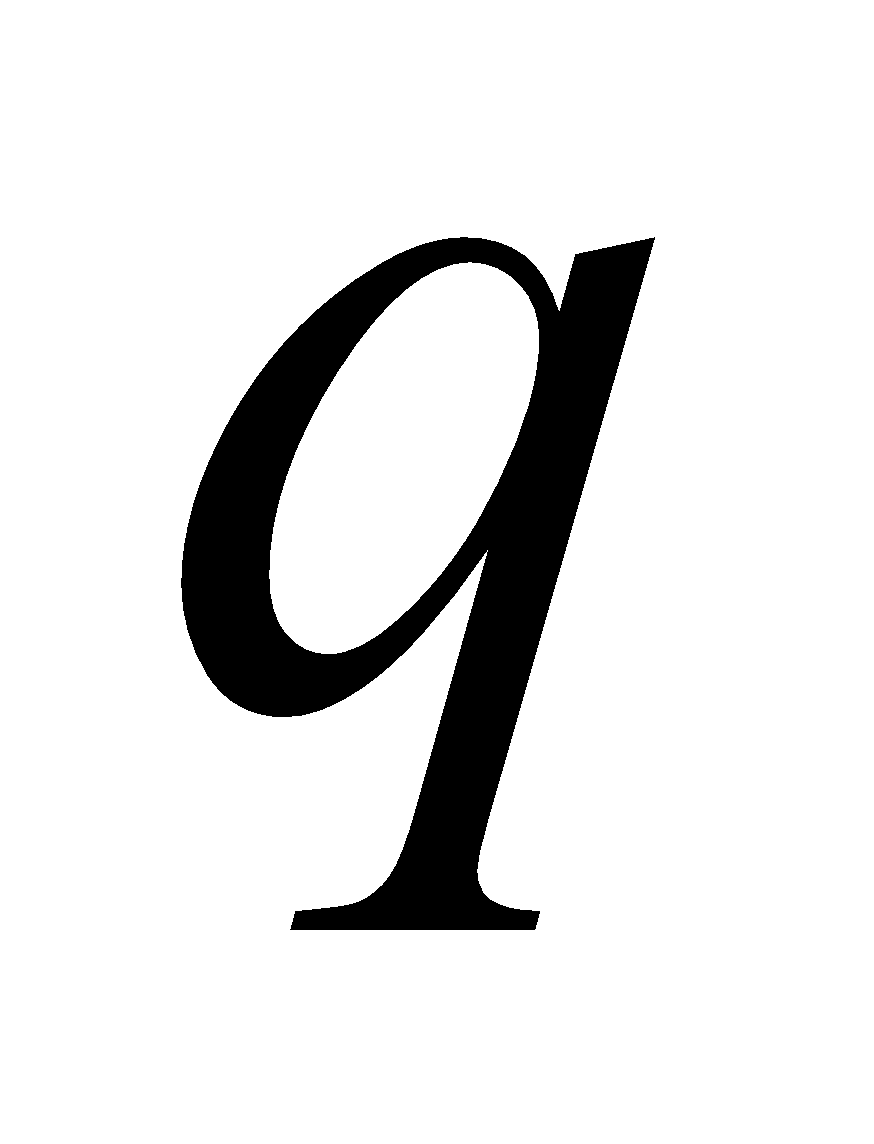
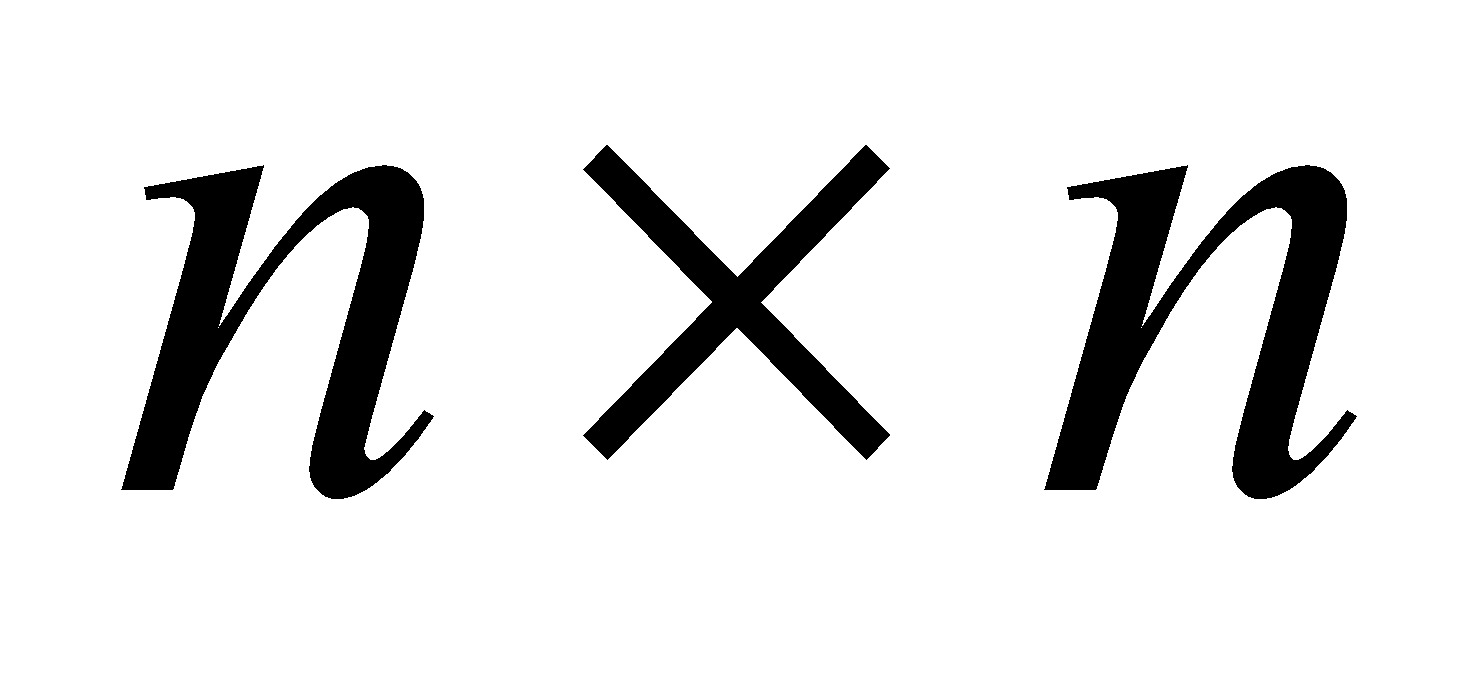
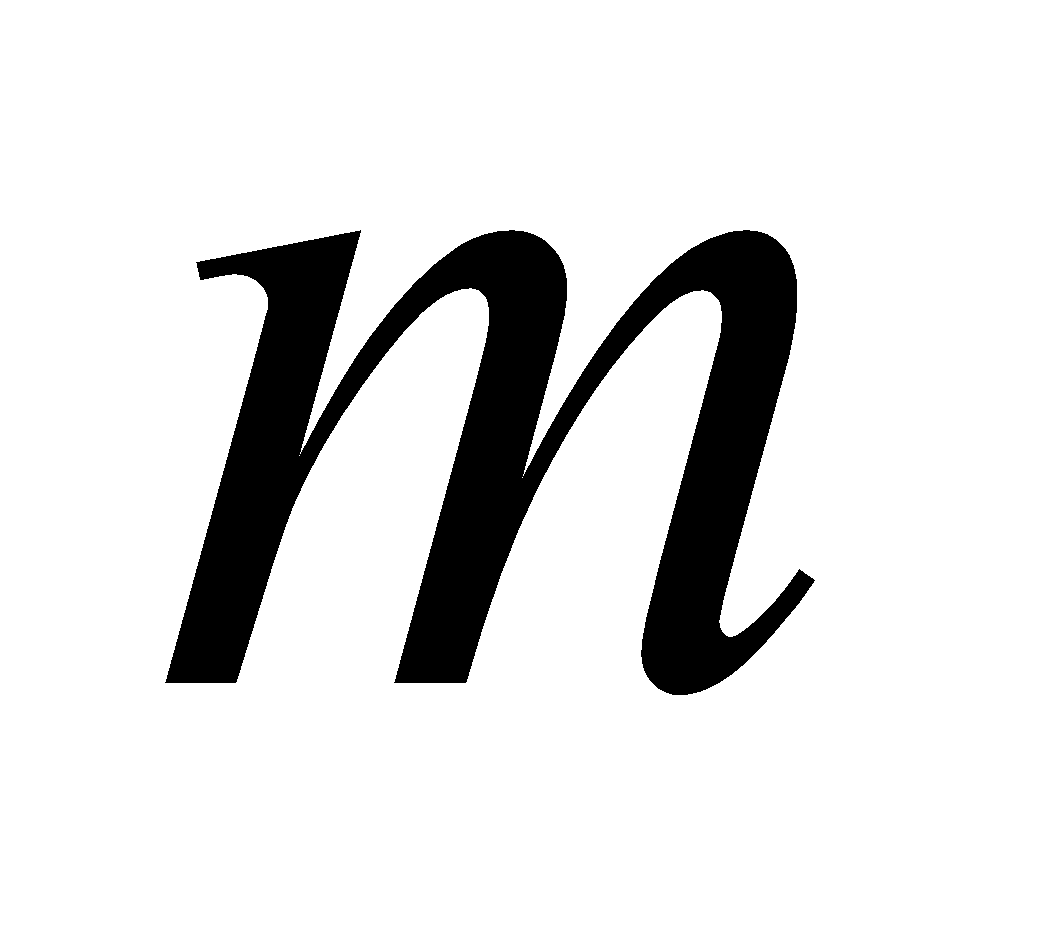
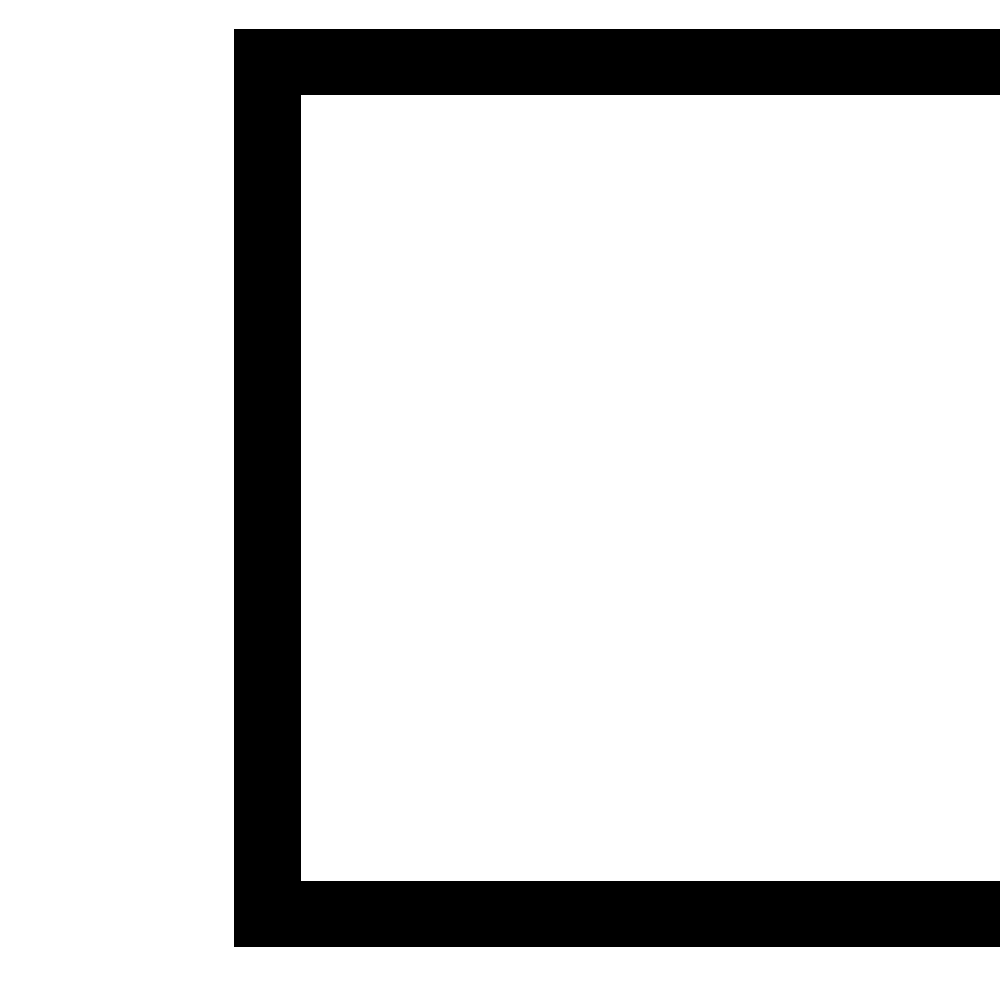
Проект “Альтернативные решения сложного пазла Lights Out” нацелен на развитие классического исследования пазла Lights Out. Пазл Lights Out – это пазл из лампочек, цель игры – погасить все лампочки. Исследование пазла Lights Out заключается в поиске решения для квадратной таблицы 3х3 с двумя состояниями: включено (1) и выключено (0). Игра начинается с квадратной таблицы, в ячейках которой находятся лампочки во включенном или выключенном состоянии. Игроки могут менять состояние каждой лампочки, нажимая на неё, однако при нажатии меняется состояние как нажатой лампочки, так и лампочек в соседних ячейках, имеющих общую сторону с нажатой. Игра заканчивается, когда все лампочки выключены. Отталкиваясь от этих результатов, мы можем обобщить задачу до произвольной таблицы 𝑎×𝑏 с любыми состояниями на любой поверхности, где и нажатая лампочка, и соответствующие ей соседние лампочки меняют состояние. Целью является обобщение пазла Lights Out и нахождение решения за наименьшее число шагов. В свете этого, мы начали свою работу с изучения имевшихся данных о соотношениях объектов в задаче, используя информацию из предыдущих исследований, и определения того, какие знания необходимы для этого проекта. Затем, используя наше знание математики, мы доказали теорему и нашли необходимые решения для этого проекта. Далее с помощью этой информации и наших наблюдений мы выделили различные случаи. Кроме того, мы написали программу, проверяющую наши методы поиска решения, для подтверждения работоспособности этих методов, для упрощения экспериментов и для лучшего понимания задачи интересующимися людьми.

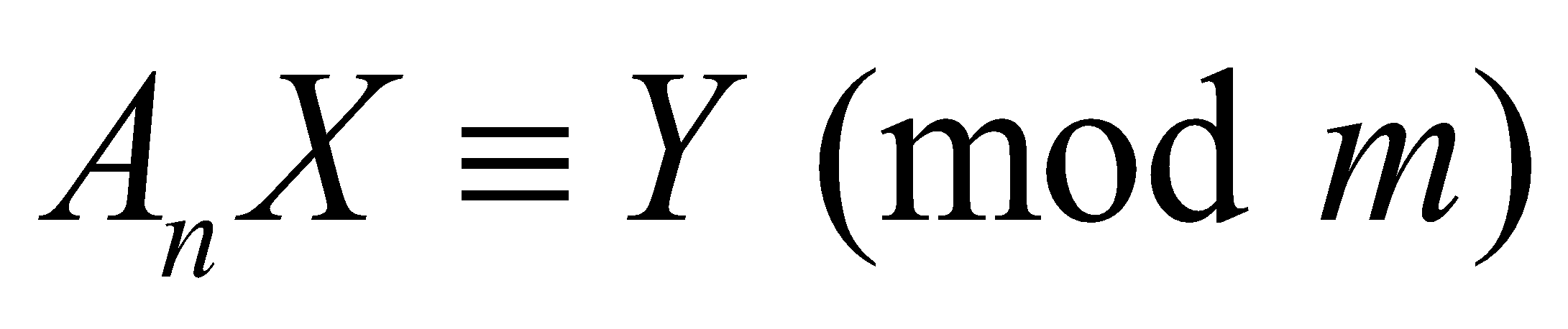
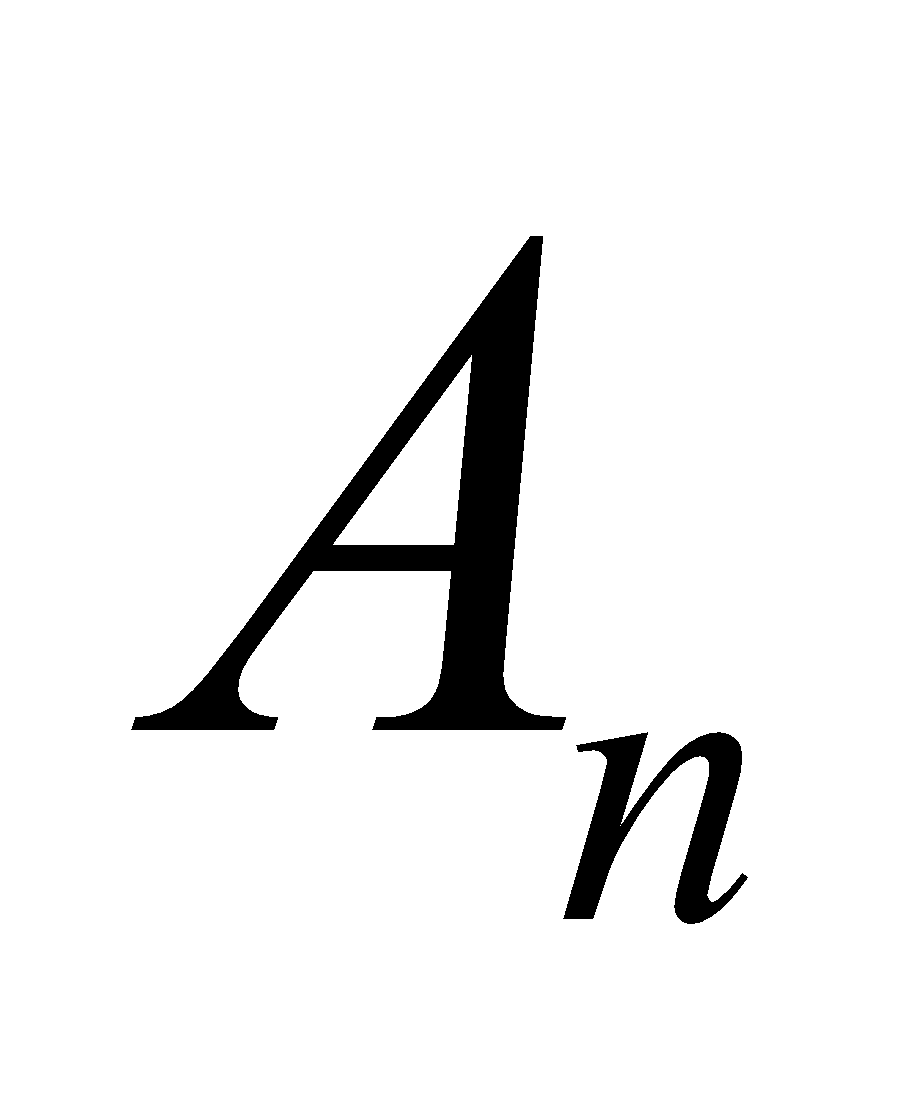
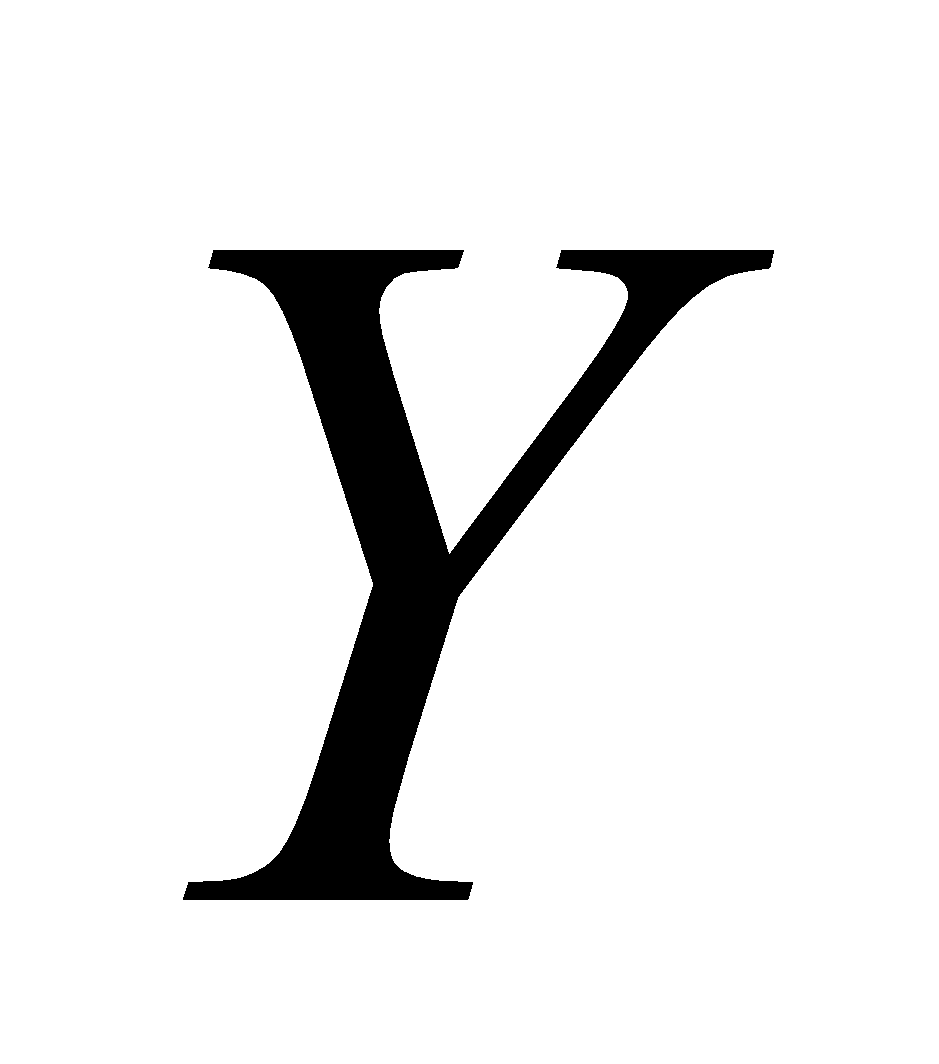
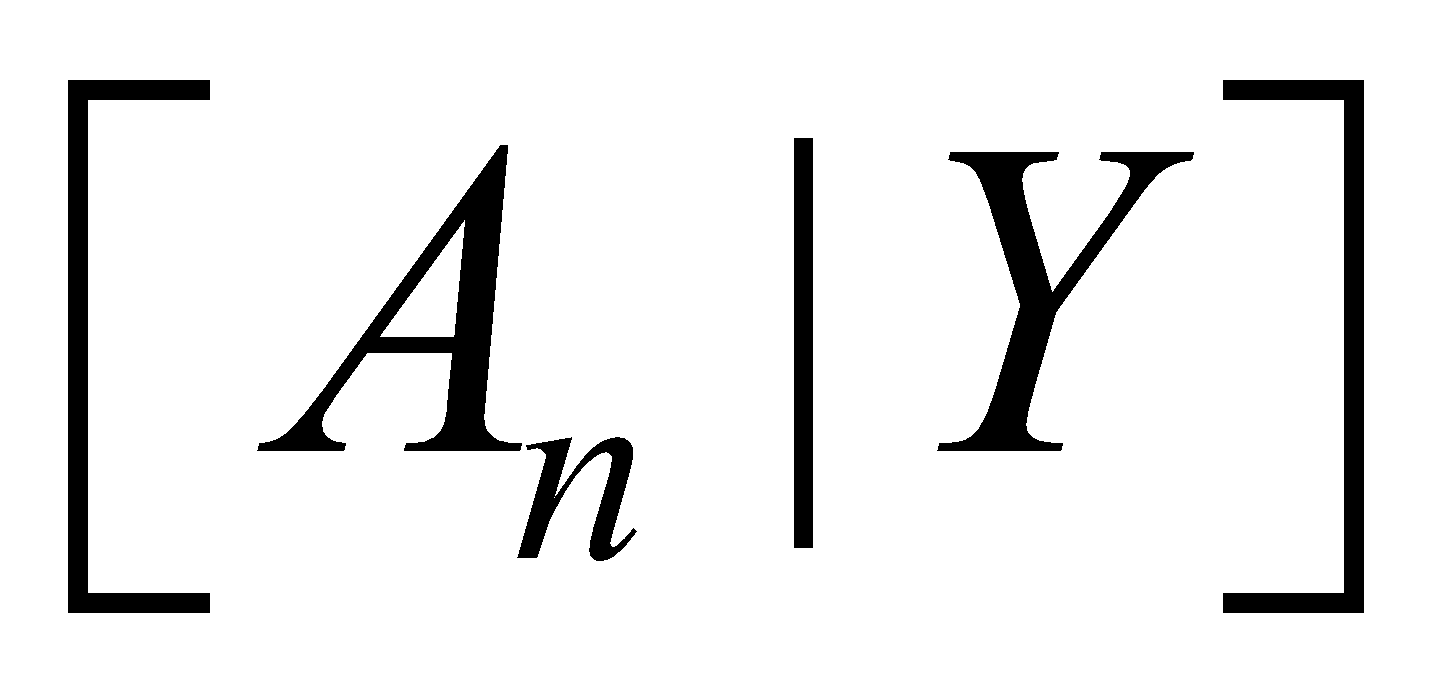
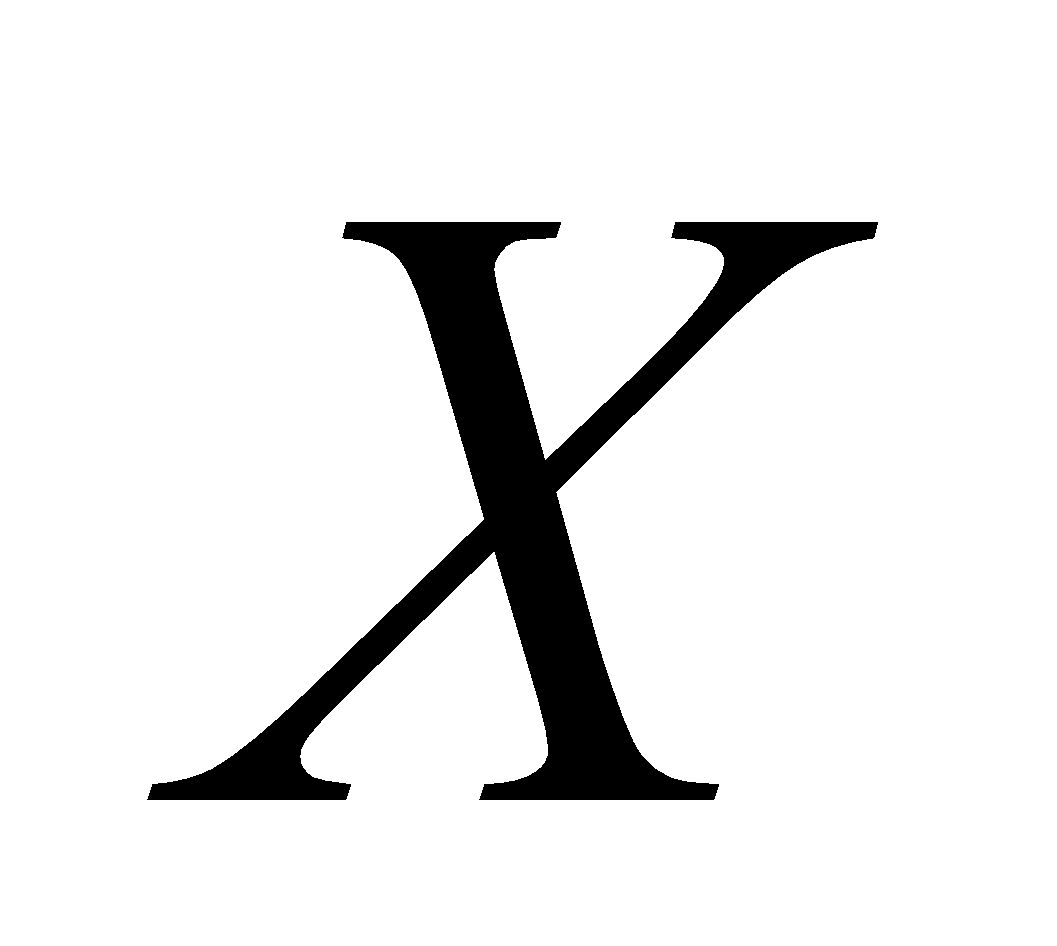
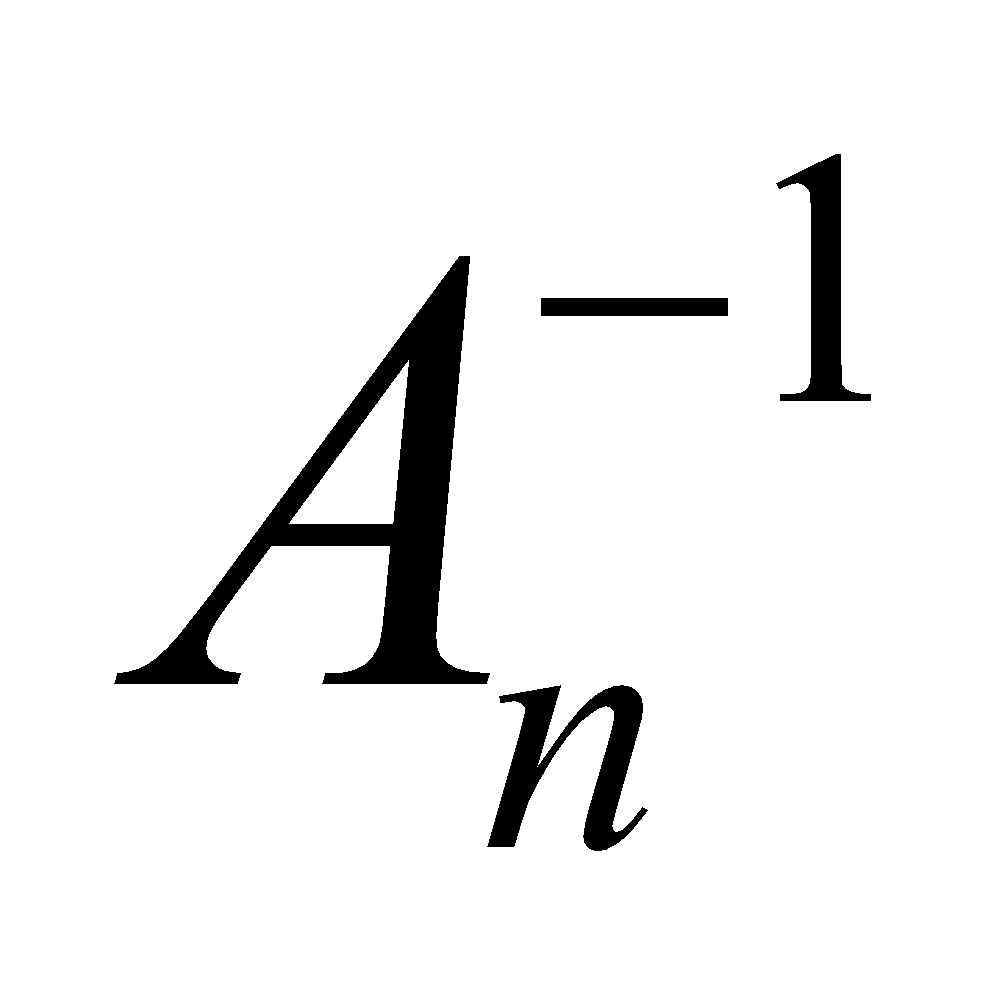
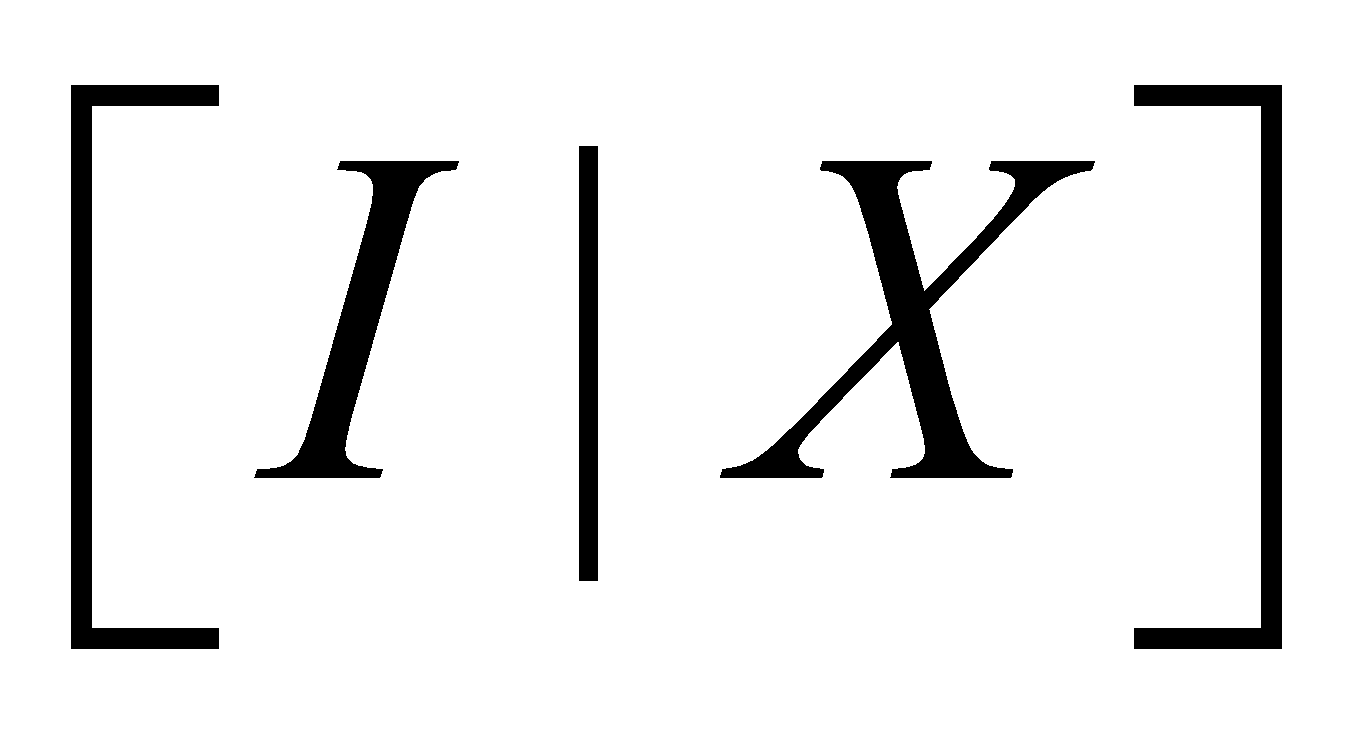
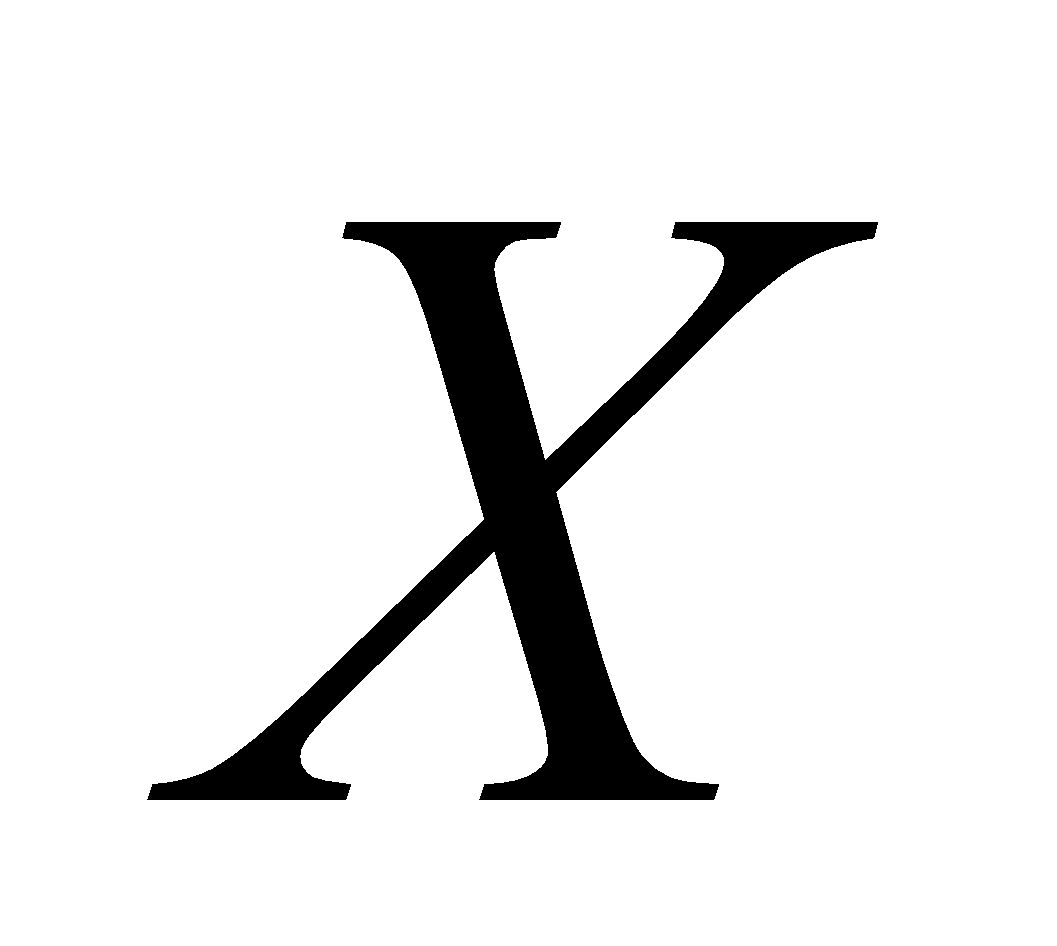
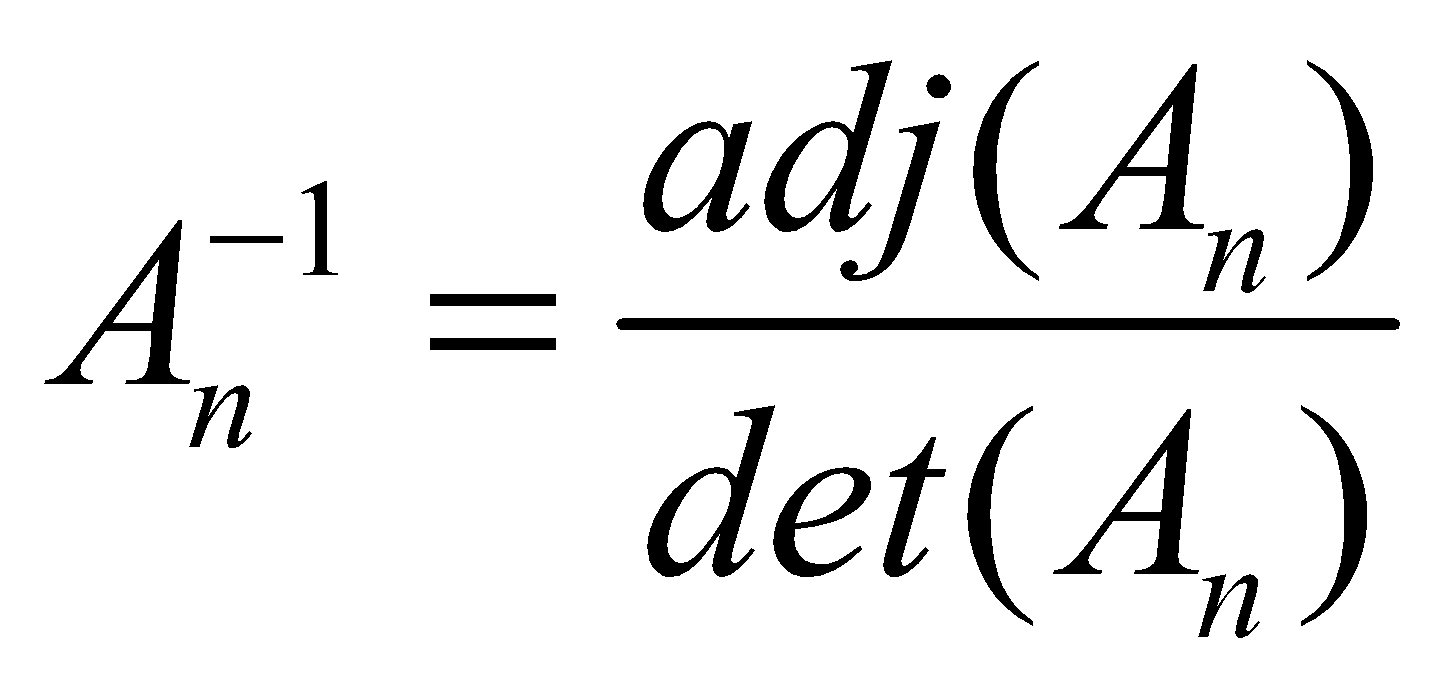
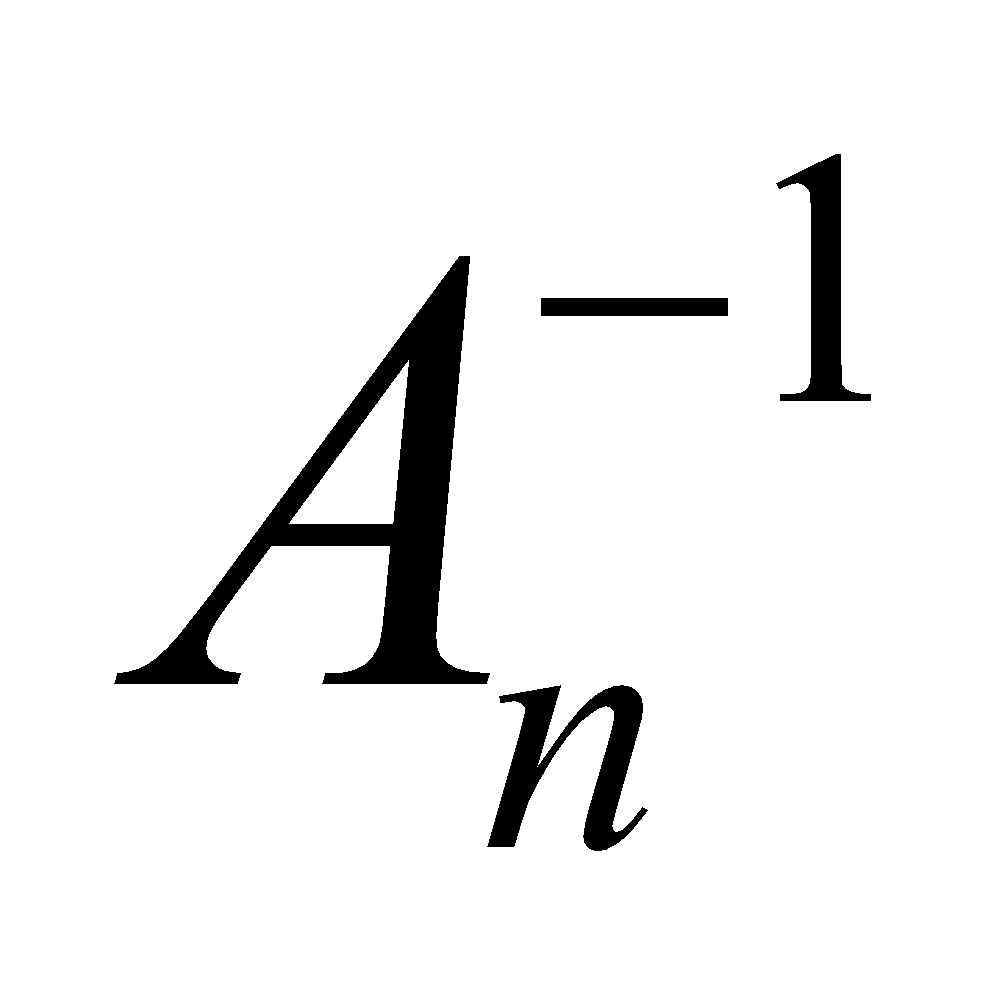
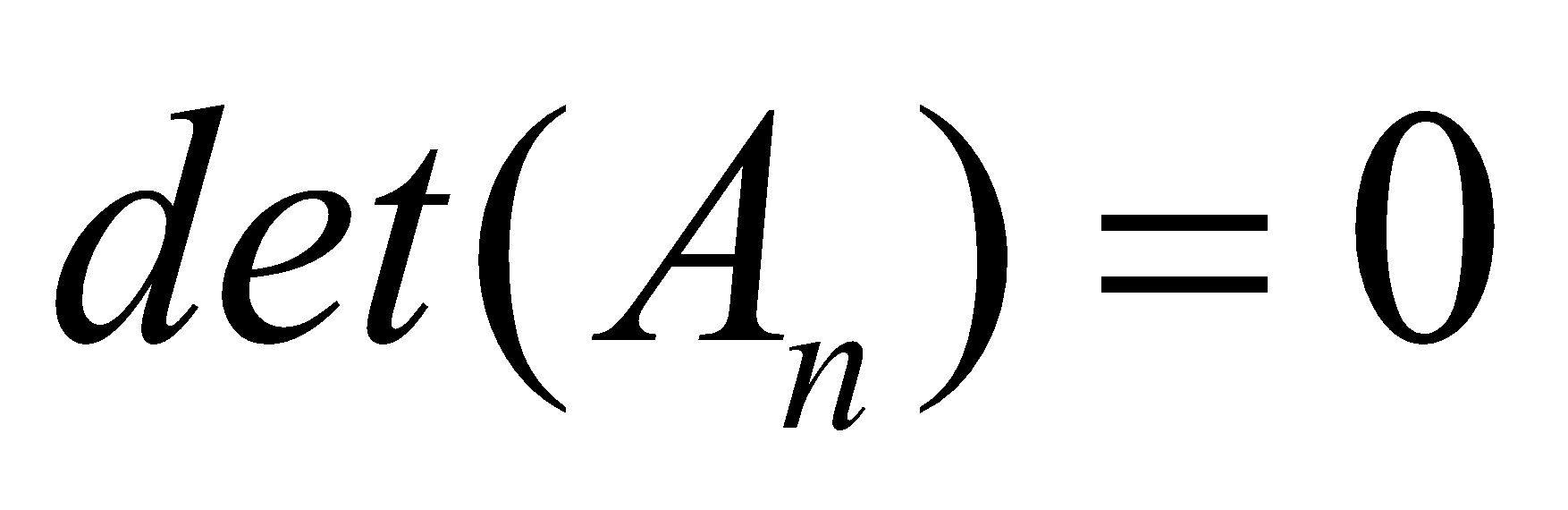
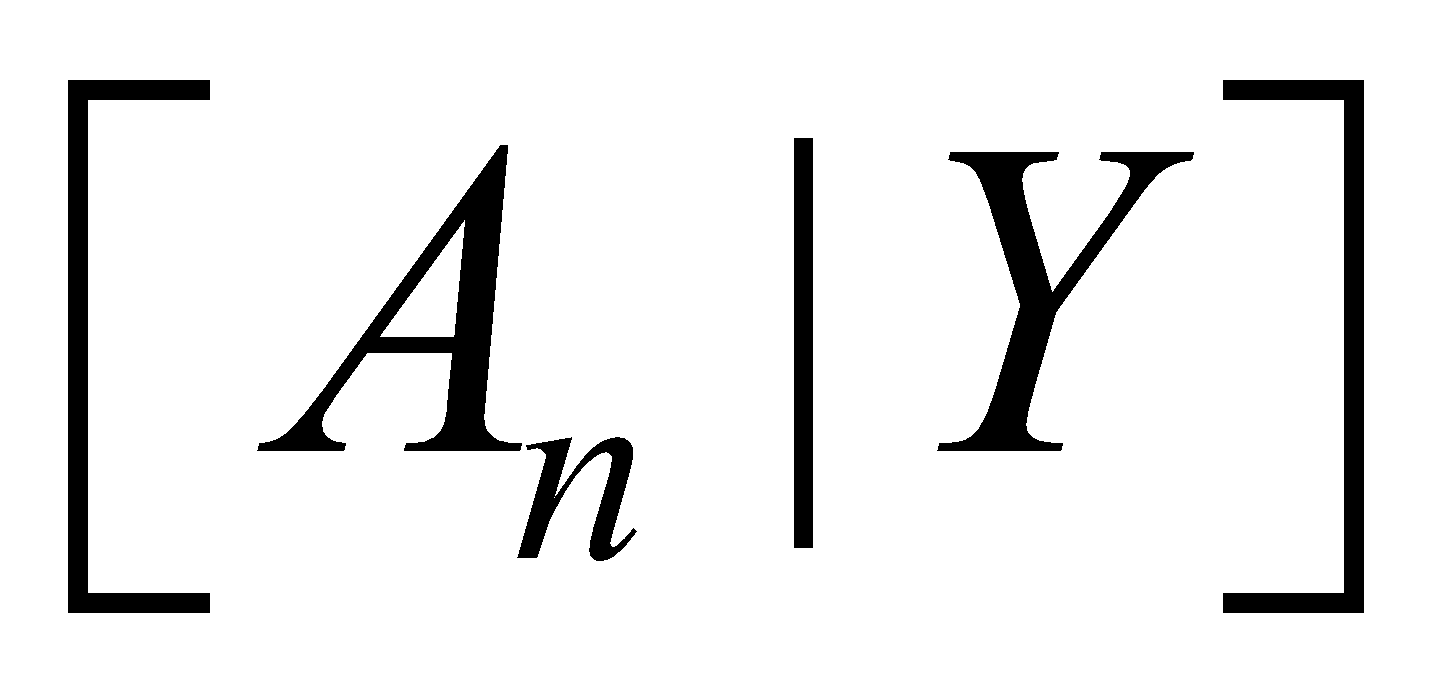
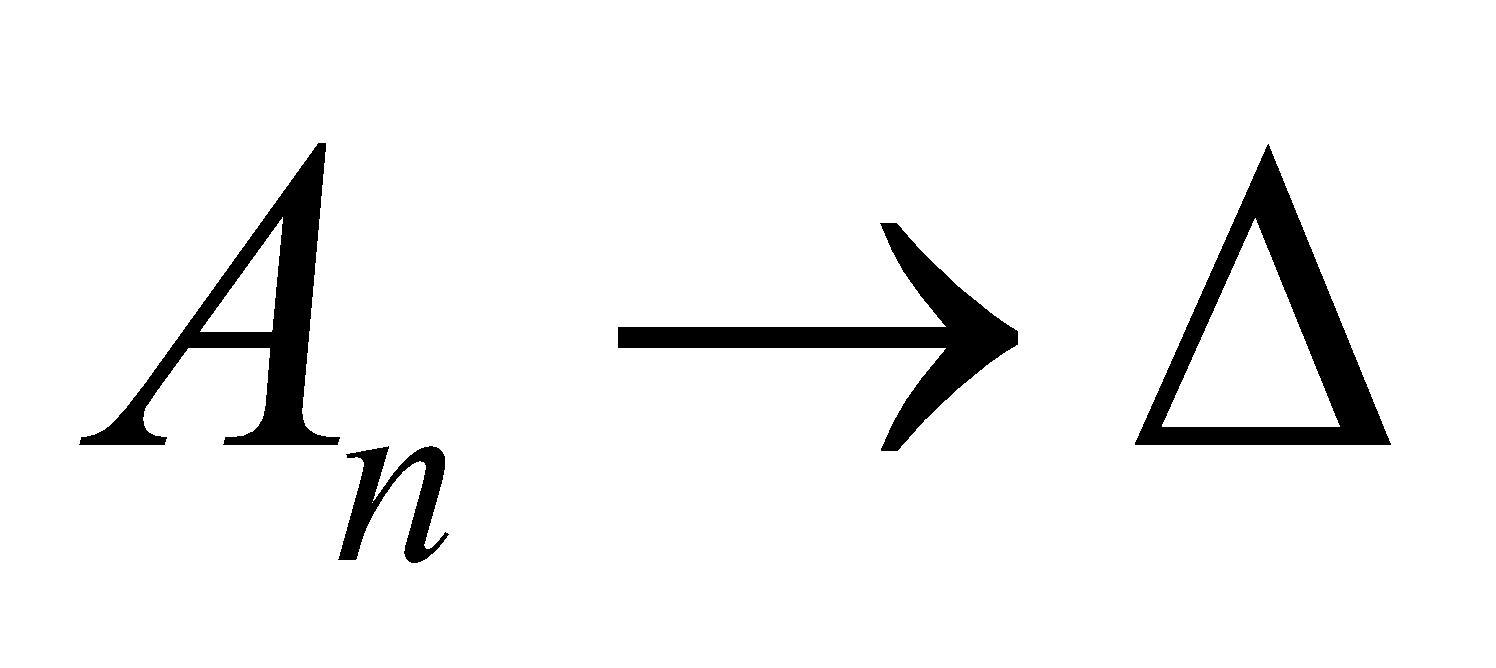
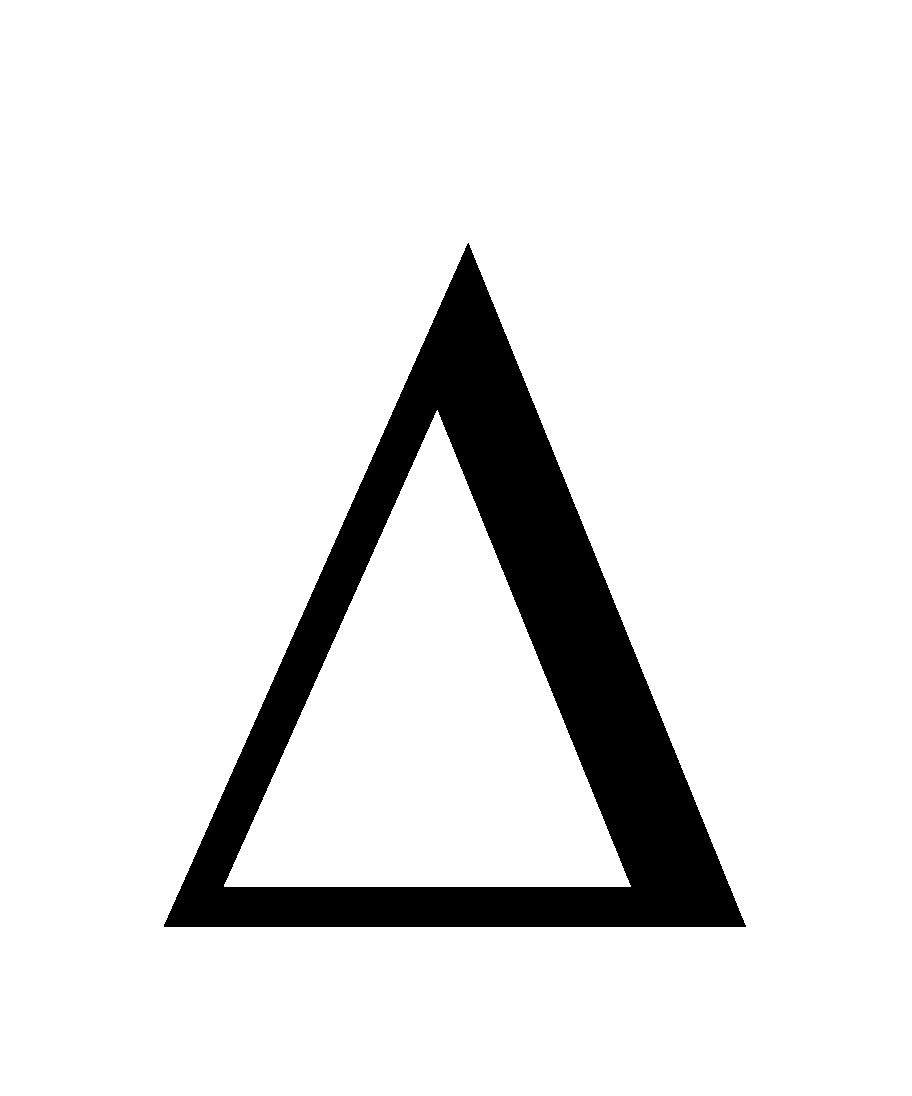
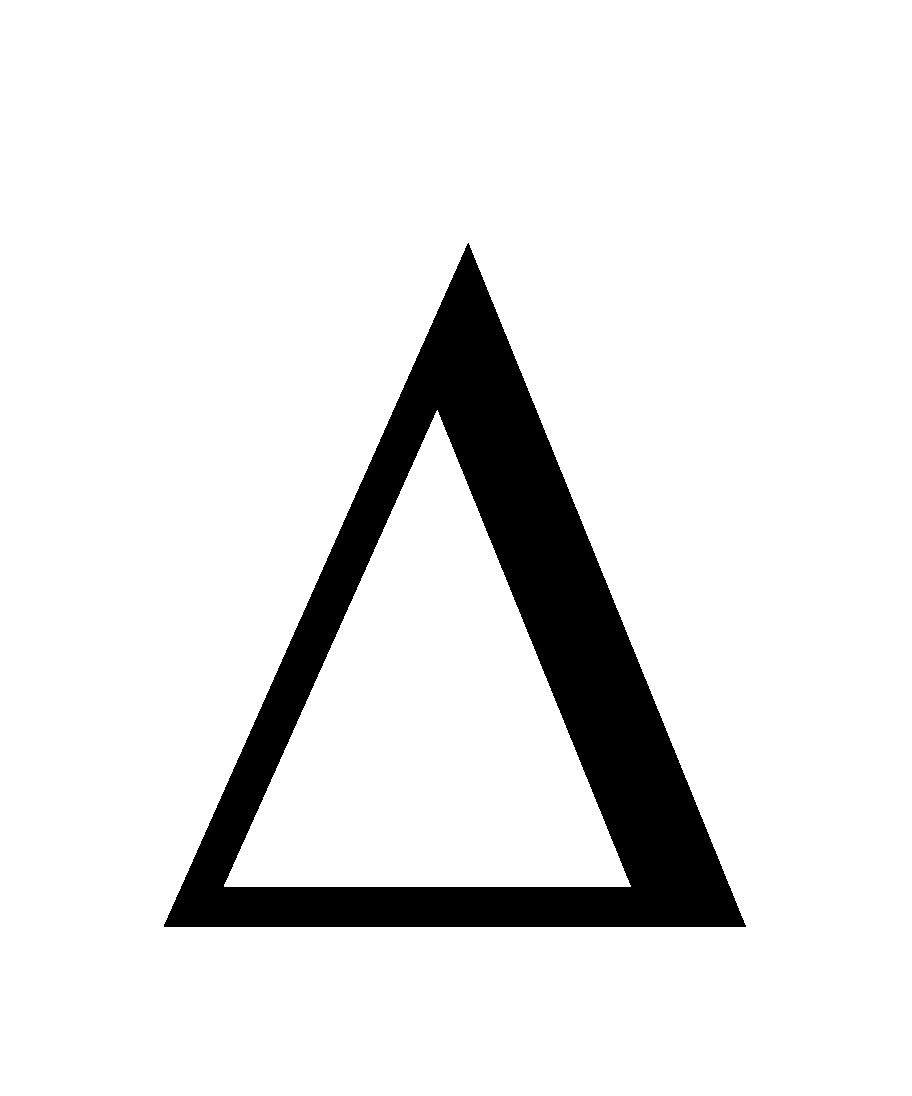
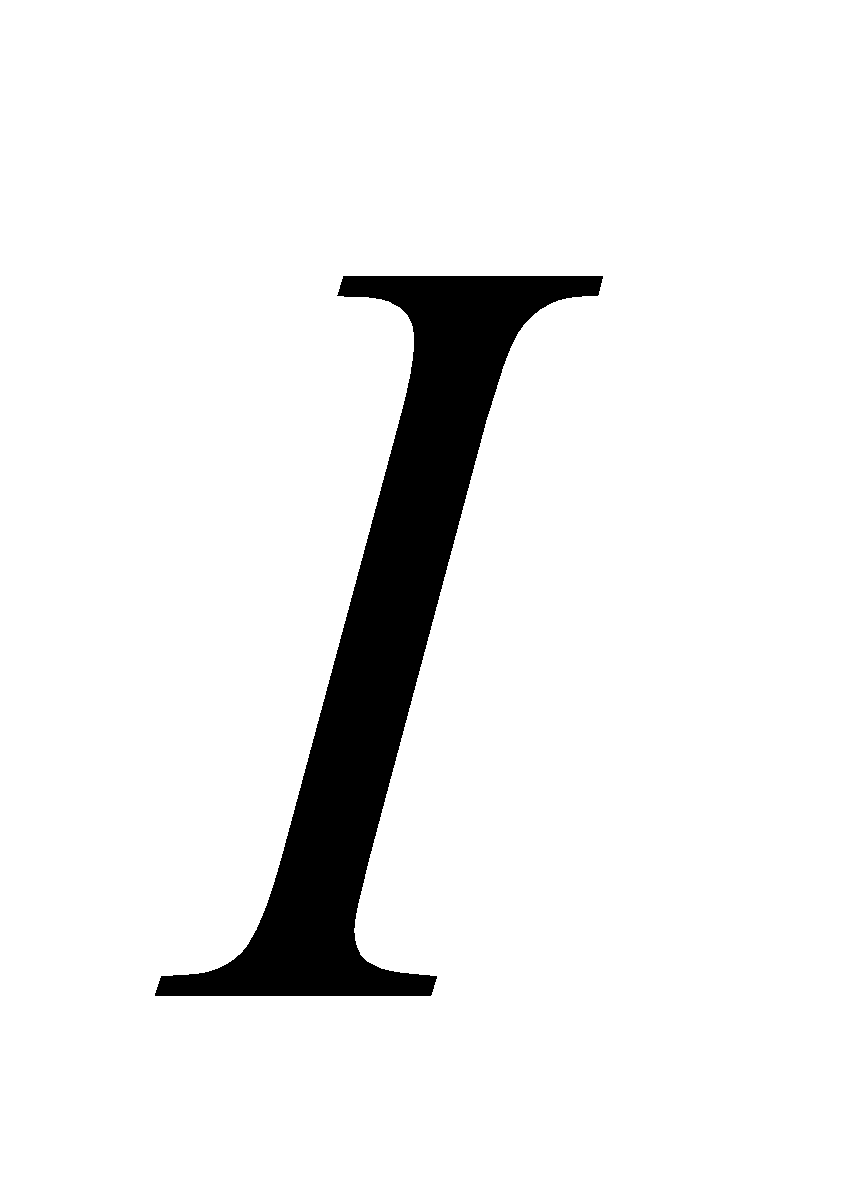
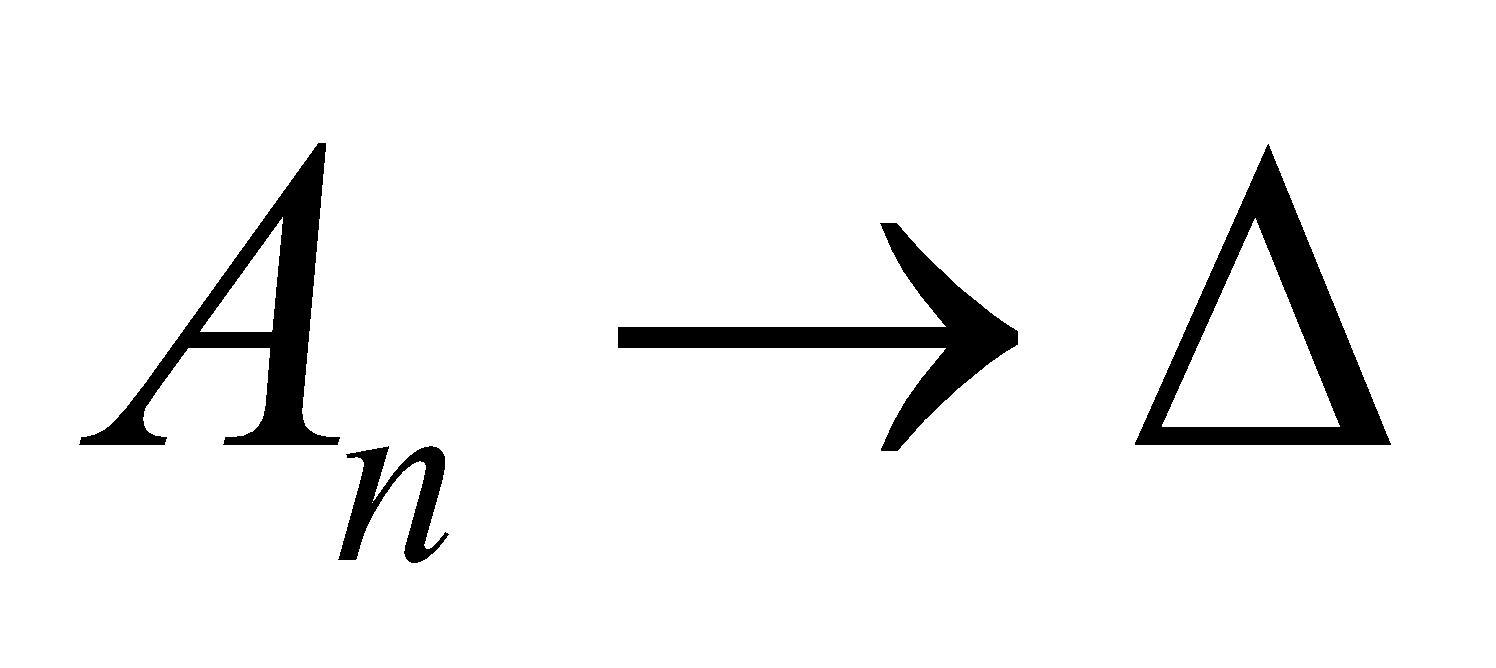
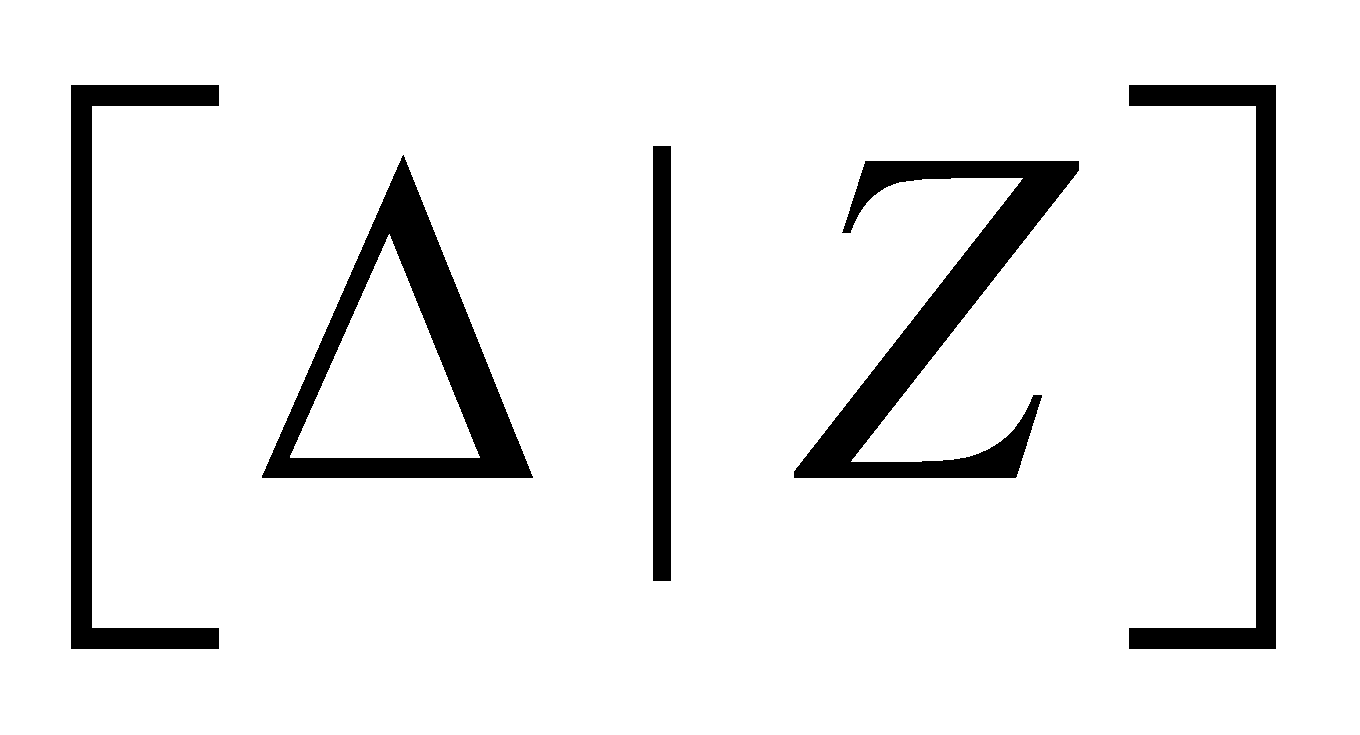
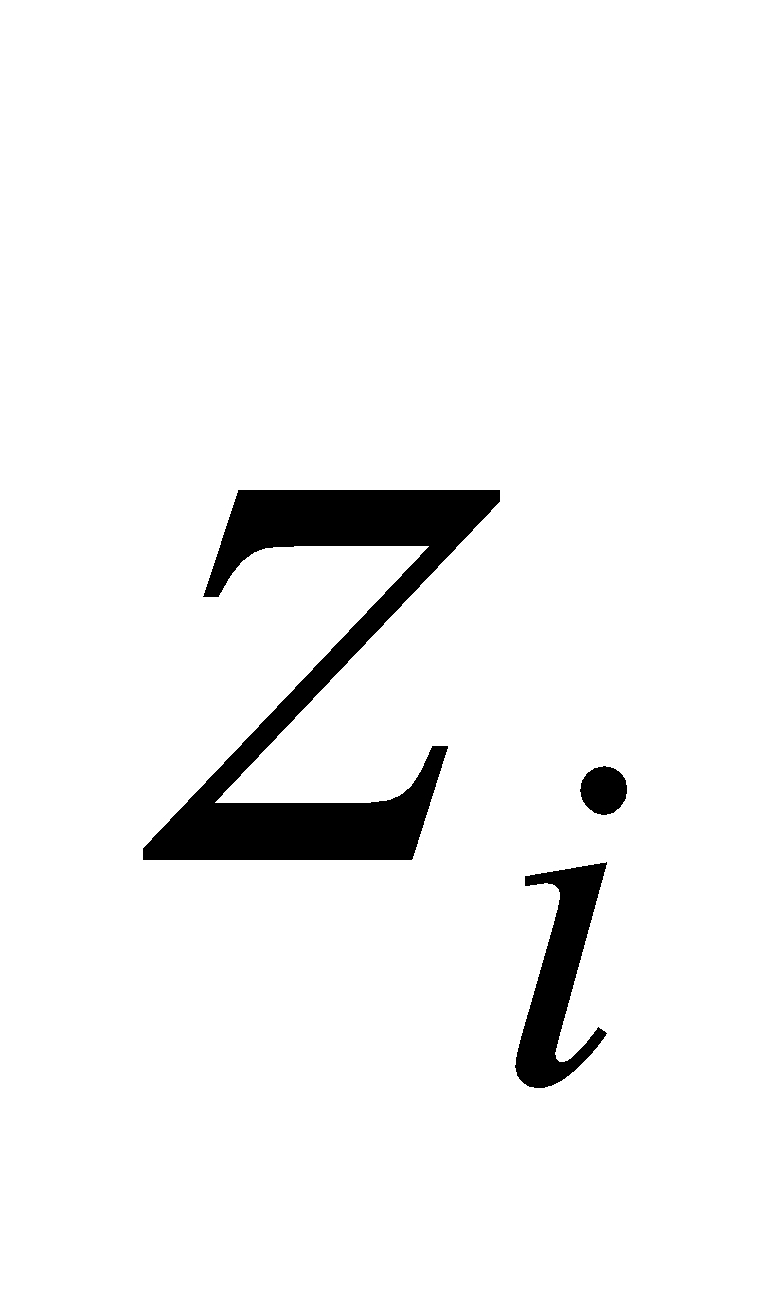
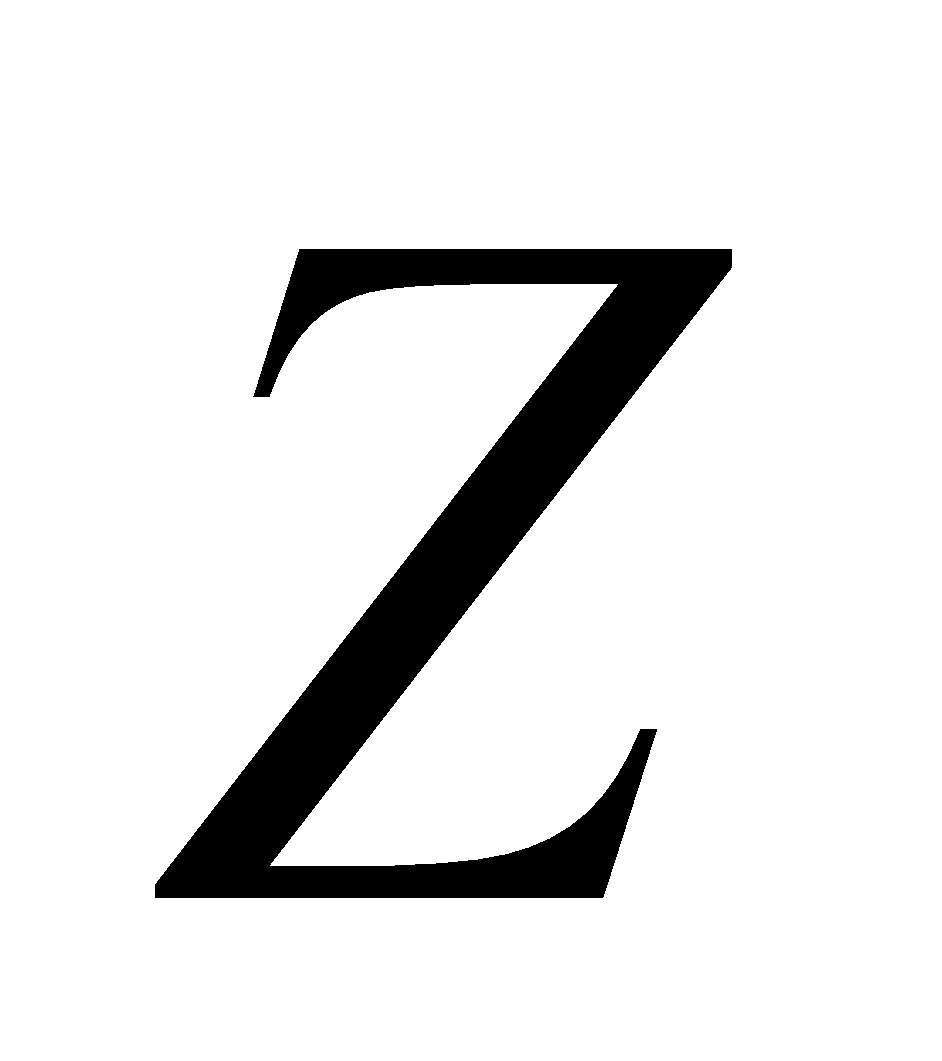
Ниже кратко представлена схема проекта. Сначала мы рассматривали только квадратные таблицы с 𝑚 состояниями. Мы можем найти решение для некоторой начальной таблицы тогда и только тогда, когда мы можем найти способ с помощью серии нажатий превратить полностью выключенную таблицу обратно в начальную. Для этого мы обозначили каждую ячейку таблицы  как i-й ящик, где i – номер ячейки в таблице.

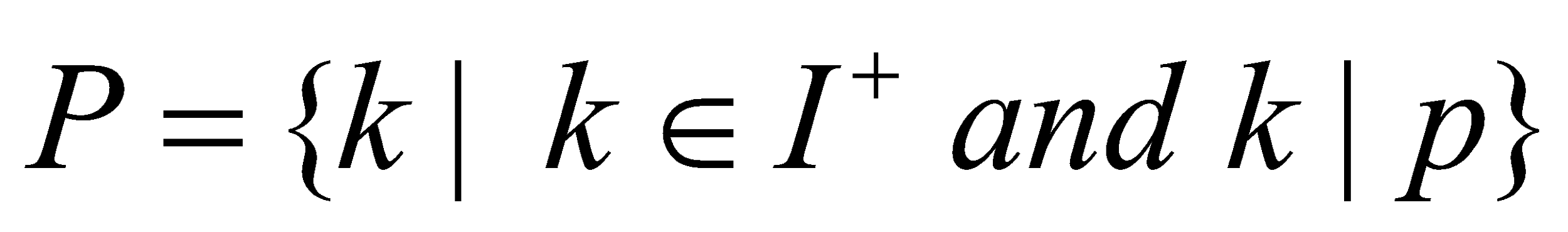
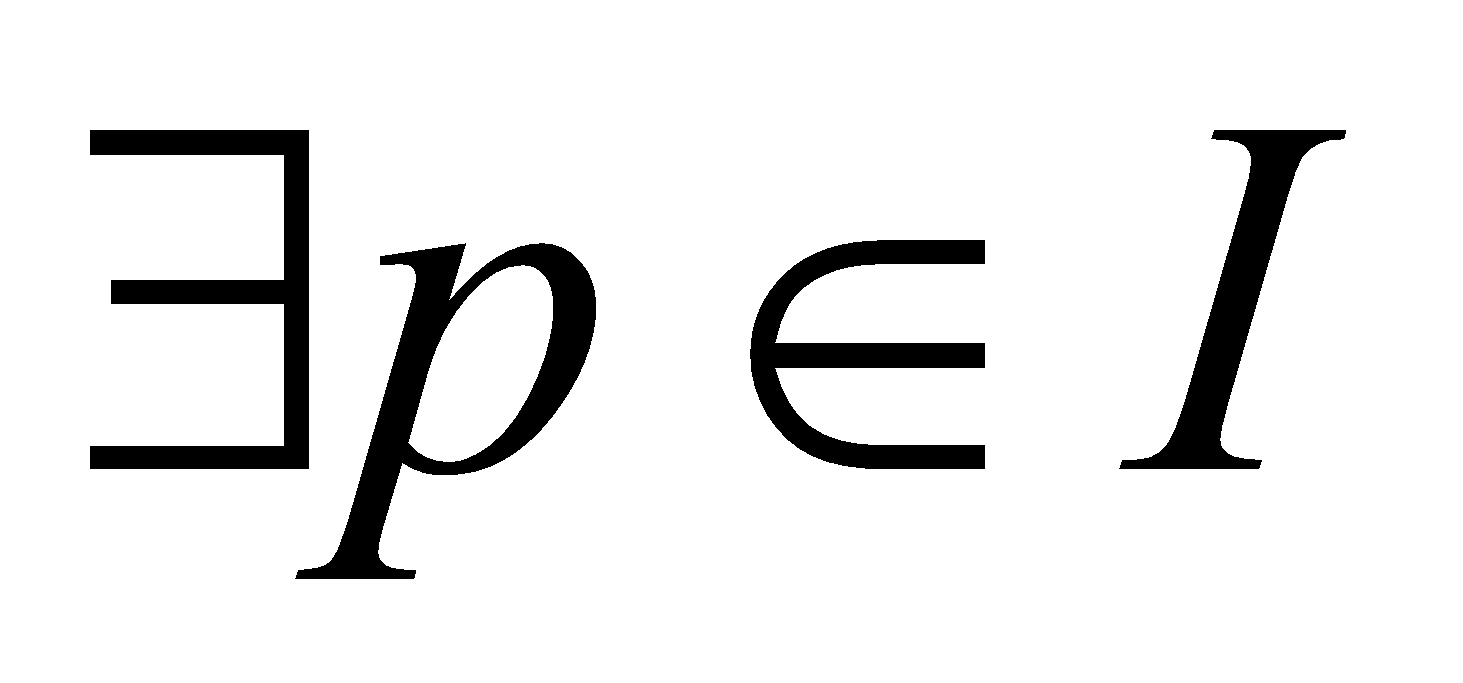
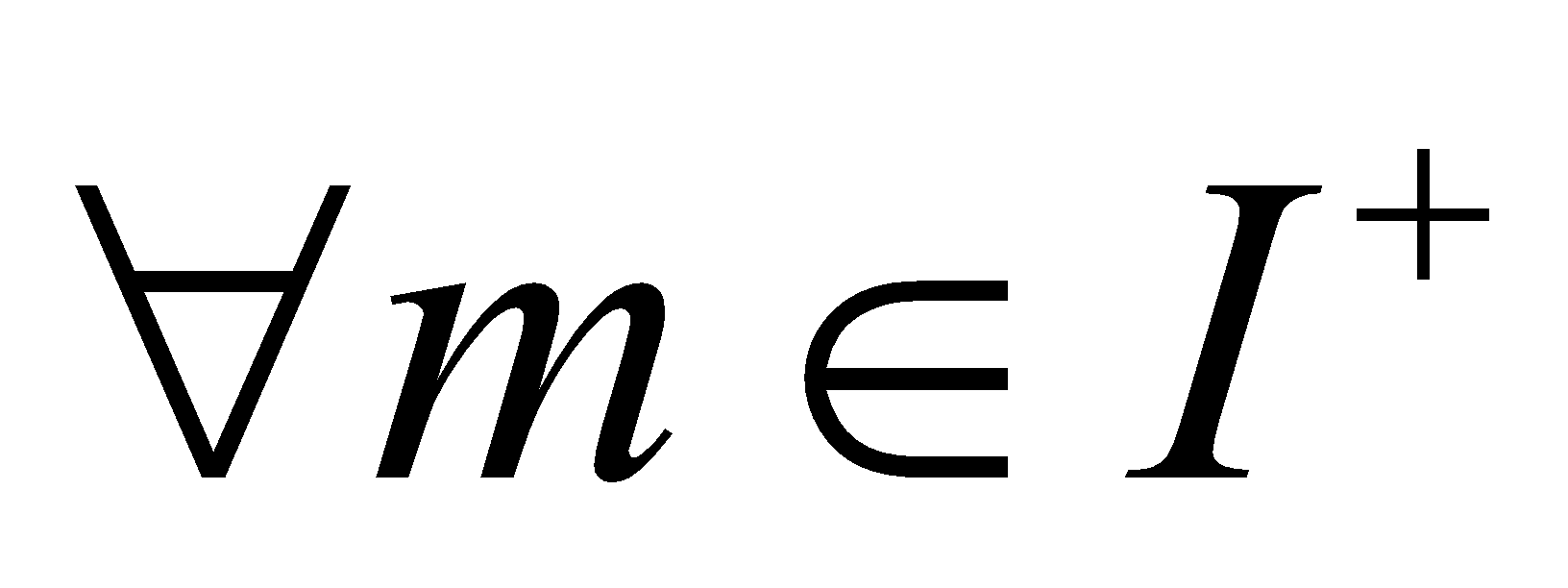
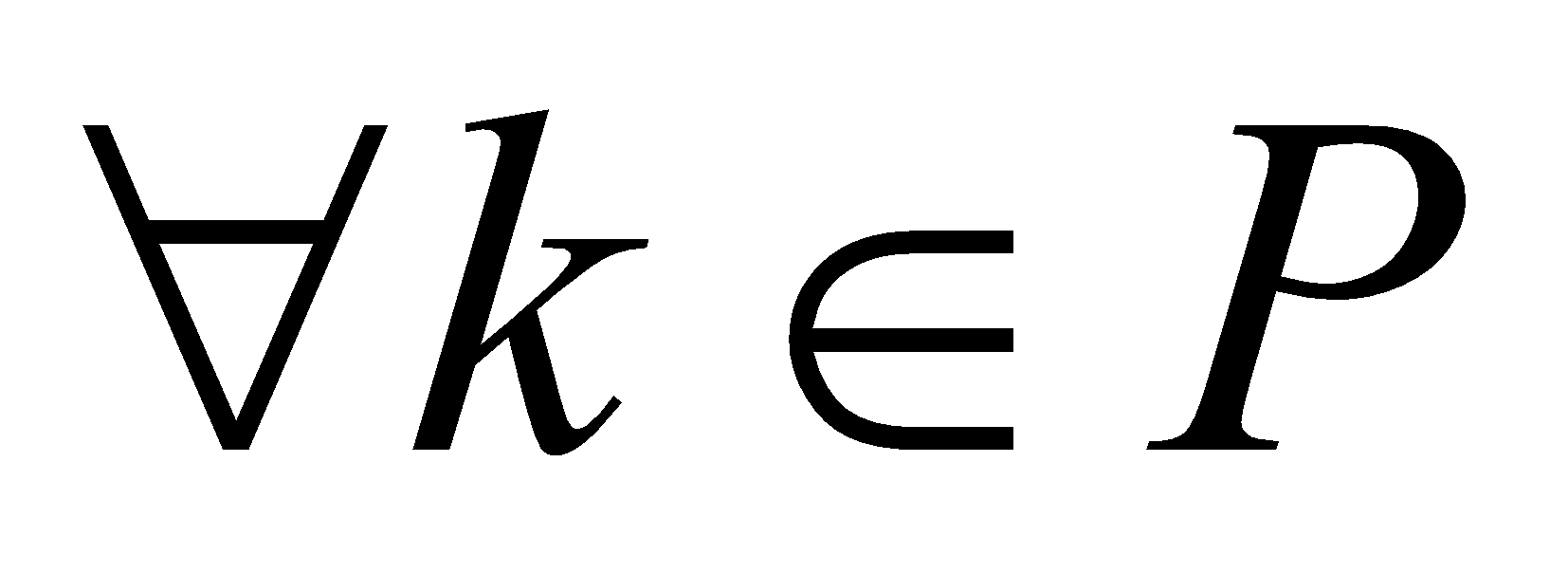
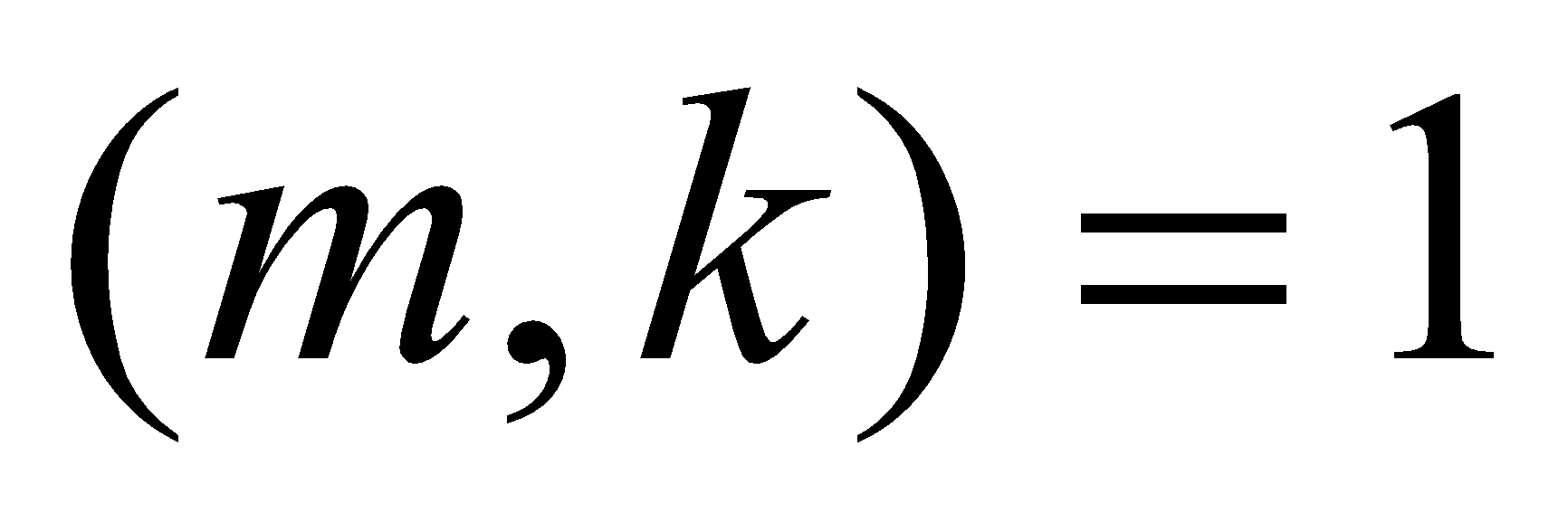
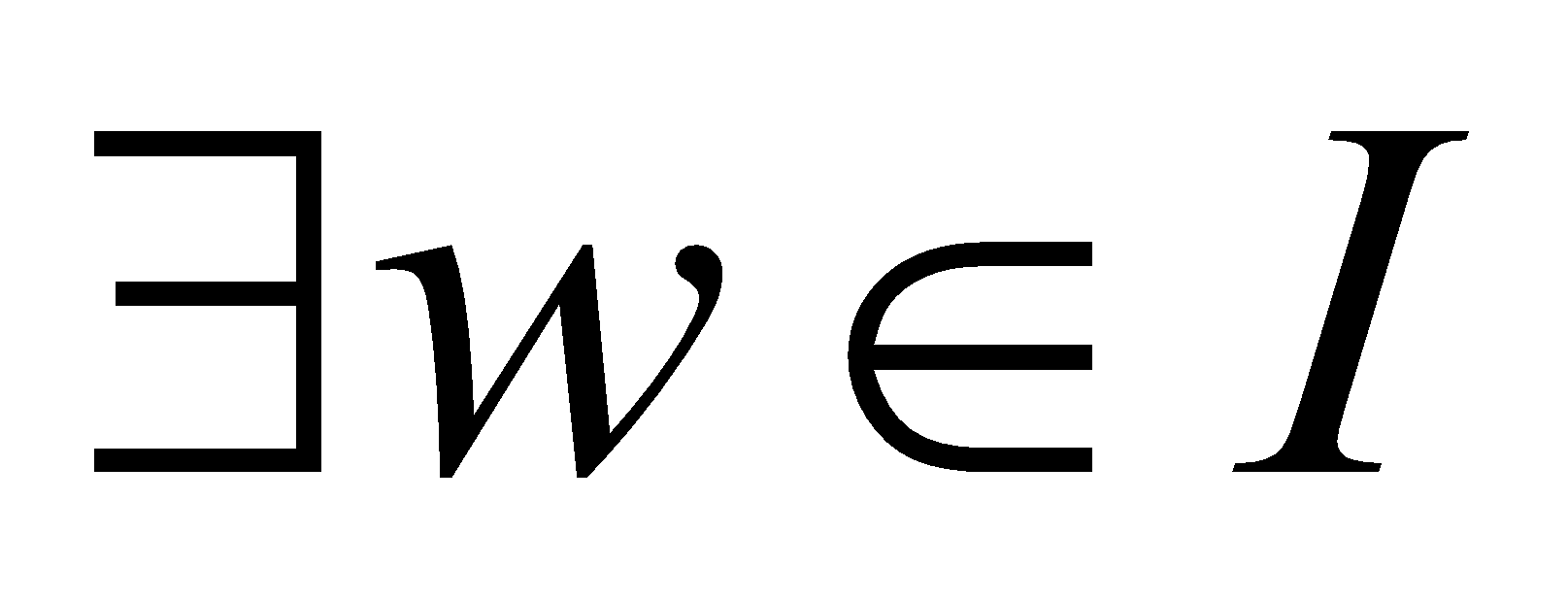
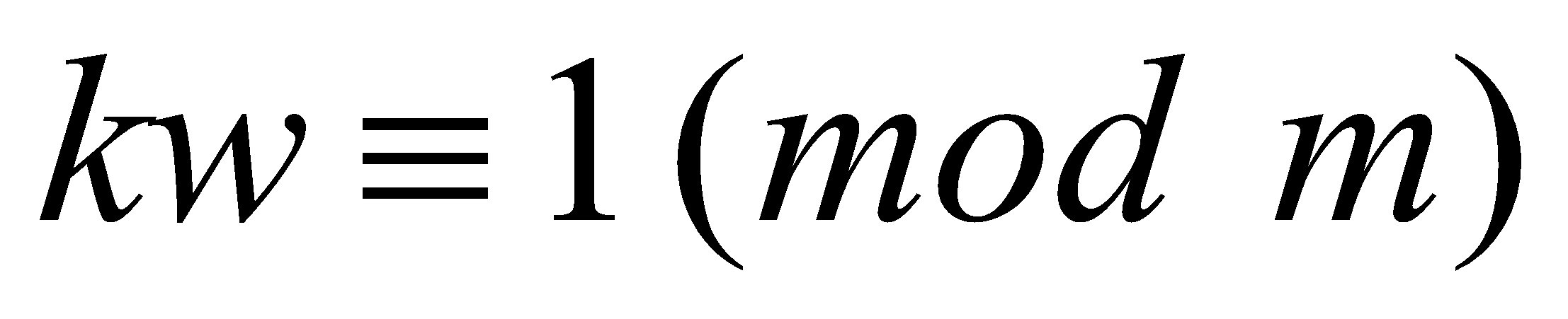
Мы определили  как матрицу размером, в которой– состояние i-го ящика, когда мы нажимаем на i-й ящик полностью выключенной таблицы. Следовательно, мы получаем уравнение , где  – количество нажатий на i-й ящик, а 𝑌 – начальная таблица. Мы можем преобразовать это уравнение, используя аппарат линейной алгебры: , где  – матрица , i-м столбцом которой является , где  – количество нажатий на i-й ящик, а 𝑖∈{1,2,3,…,𝑛2}. Поскольку ячейки таблицы могут находиться в одном из 𝑚 состояний (0,1,…,𝑚−1), мы получаем уравнение . Как следует из уравнения,  – это набор нажатий, которые переводят полностью выключенную таблицу (0) в начальную таблицу (𝑌): .

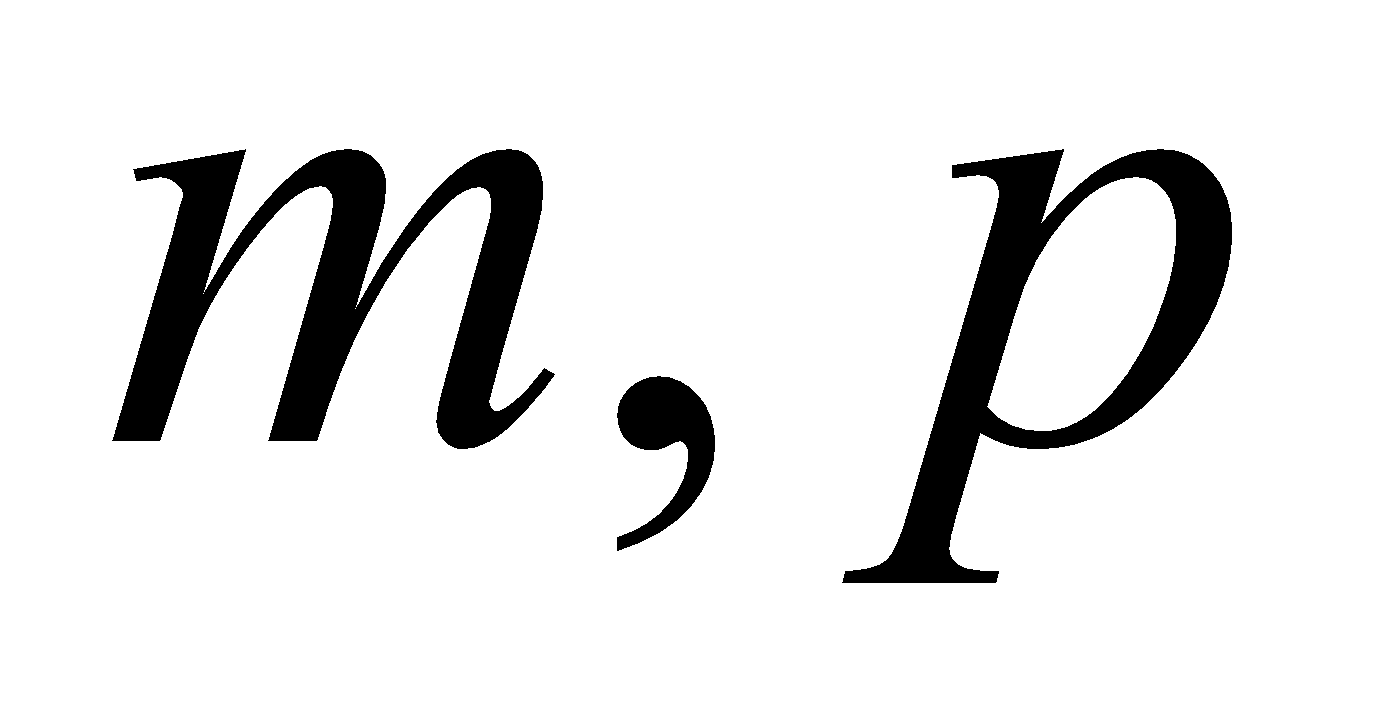
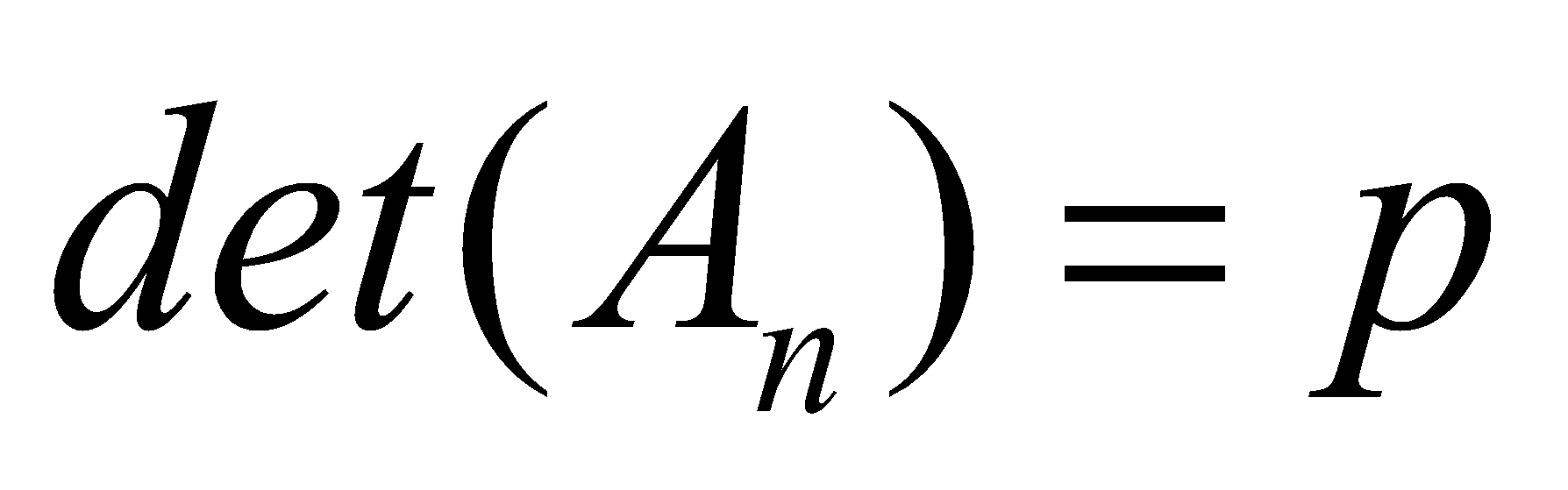
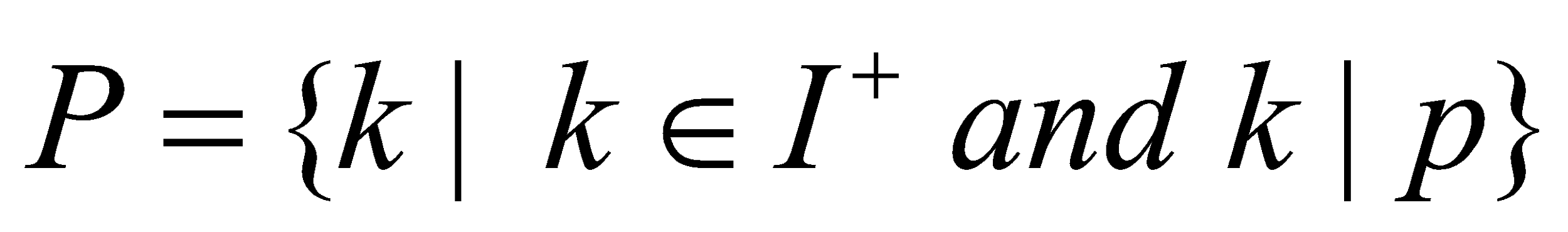
Таким образом,  является решением, т.к. это серия нажатий для получения . Поскольку существуют таблицы, в которых мы не можем найти решение , мы найдём решение ,где  таблица, в которой все лампочки находятся в состоянии , а  состояние начальной таблицы. Вообще говоря, решения могут быть либо конечными, либо бесконечными. Однако, поскольку мы рассматриваем задачу по модулю , решения обязаны быть конечными. Проделав некоторые вычисления, мы обнаружили, что количество таблиц с  состояниями, для которых можно найти решение, равно  ,а количество таблиц, для которых нельзя найти решение, равно  где  количество решений.

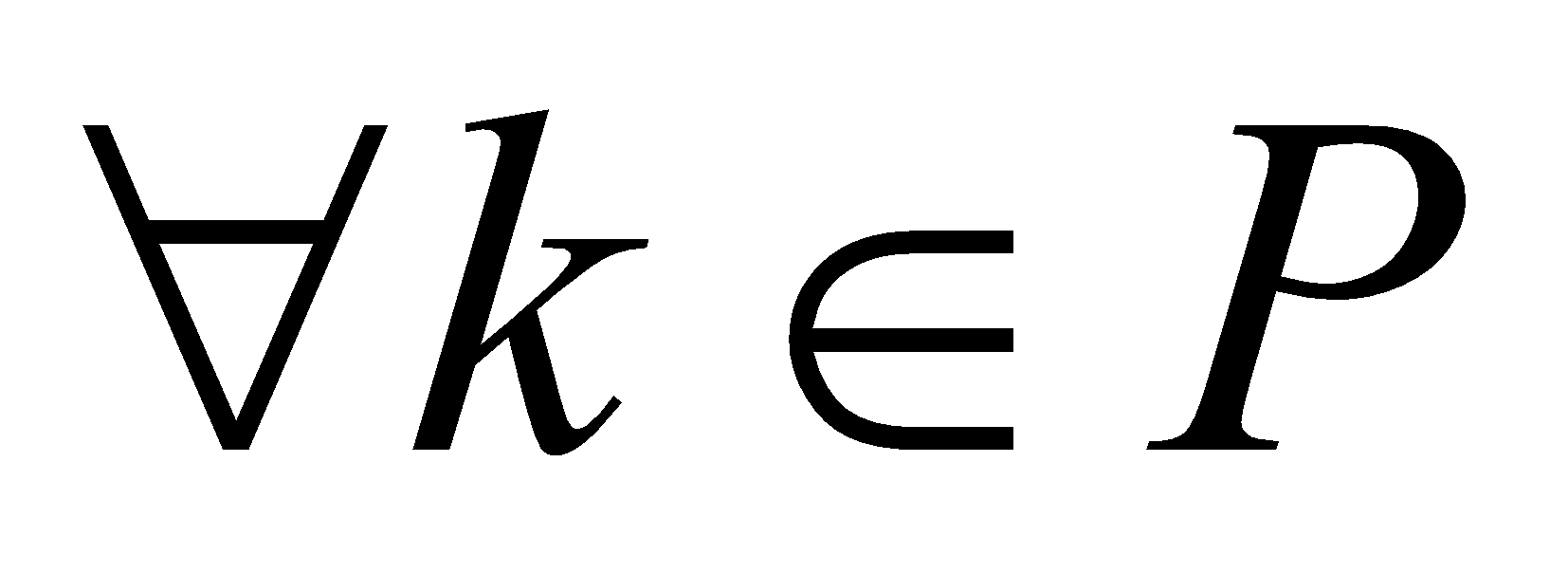
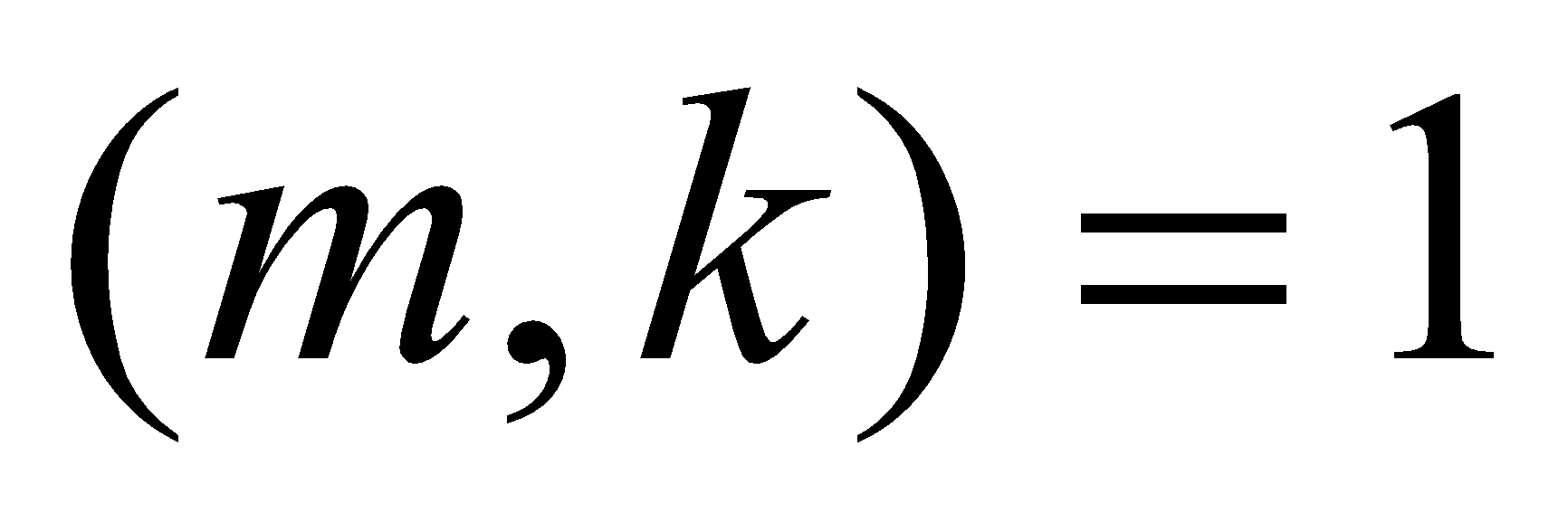
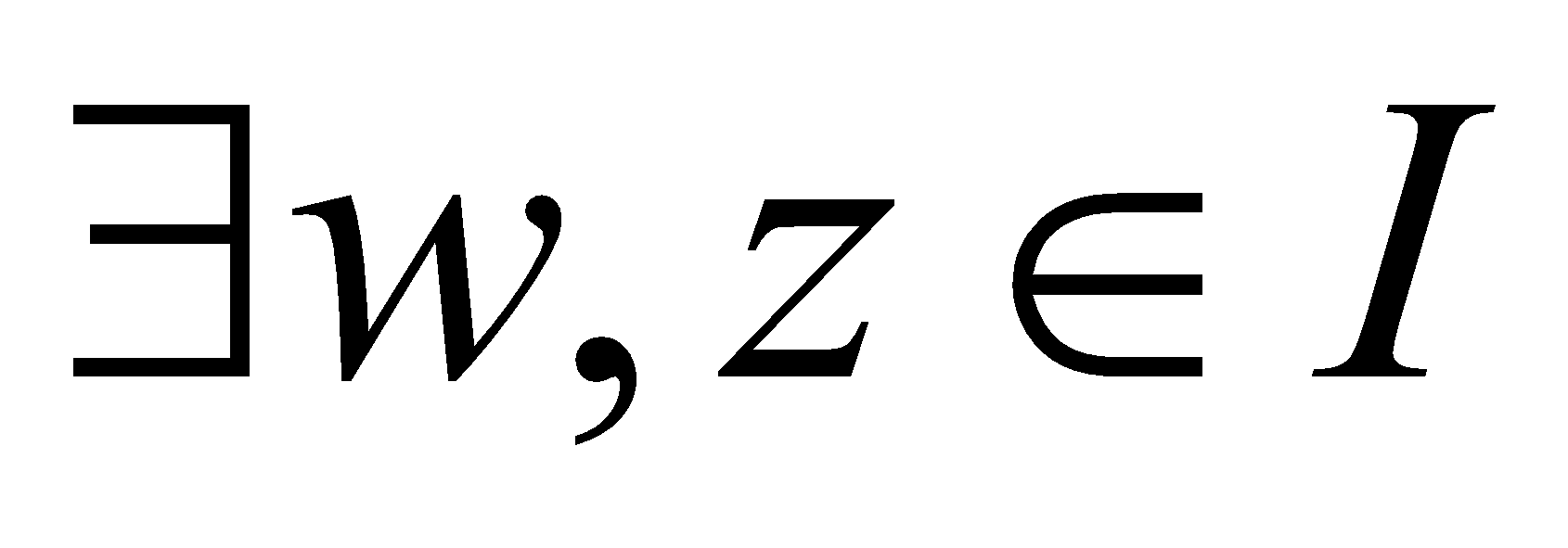
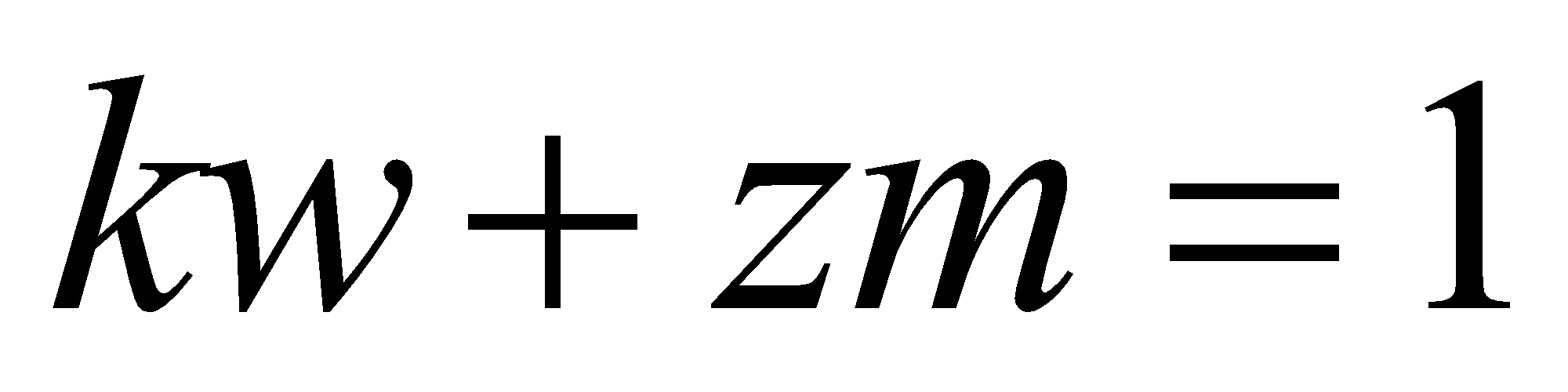
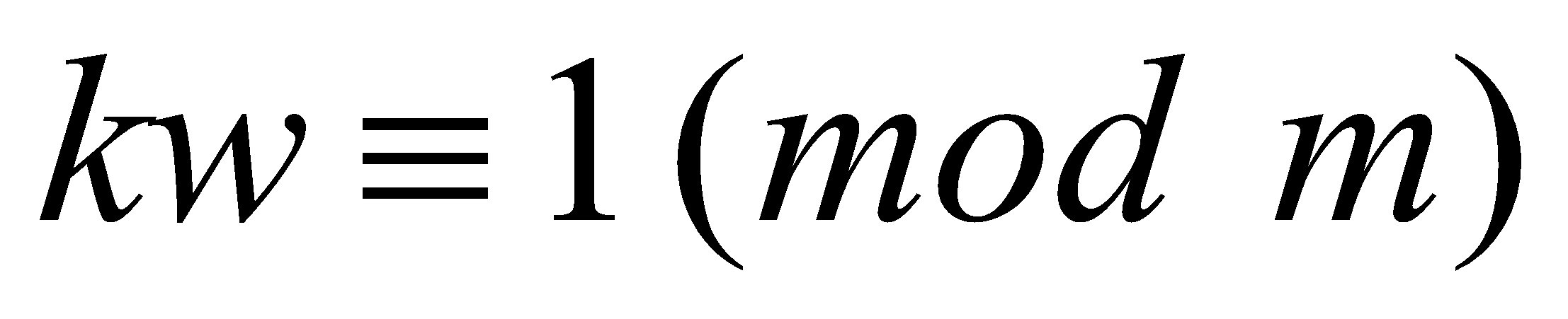
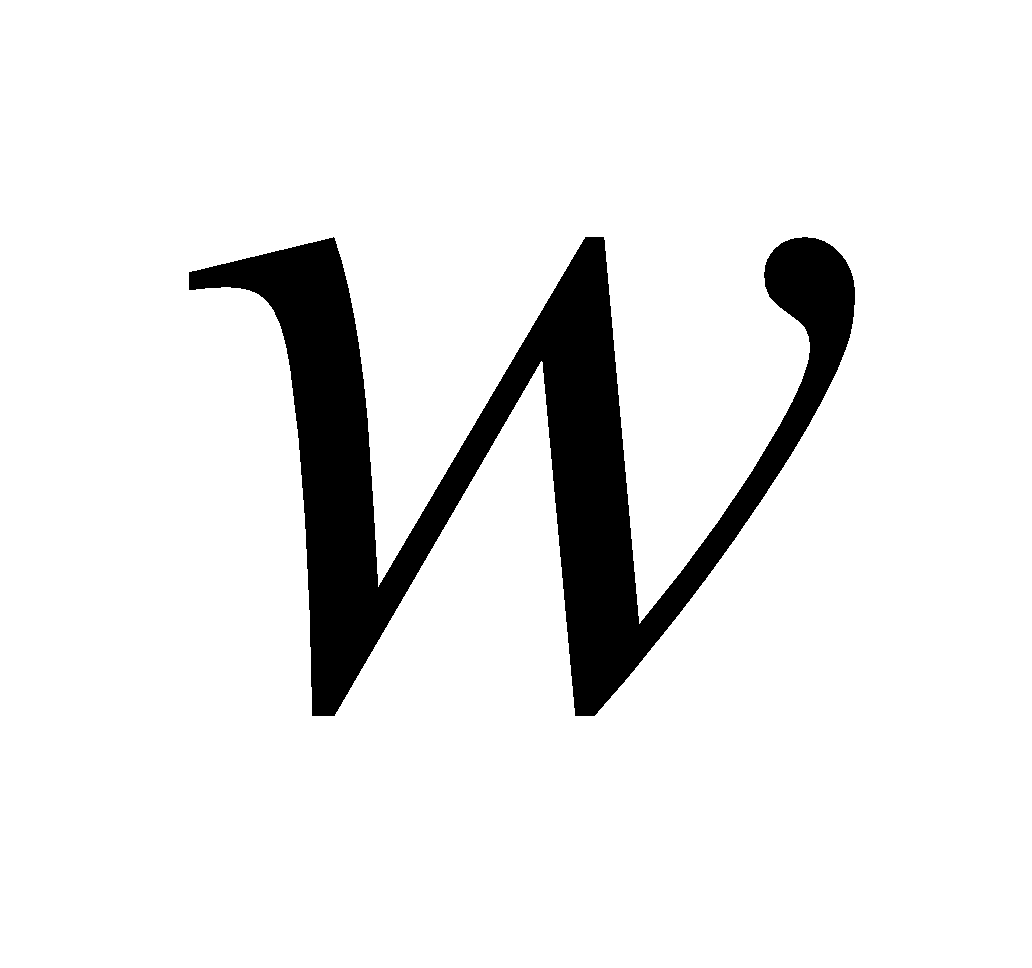
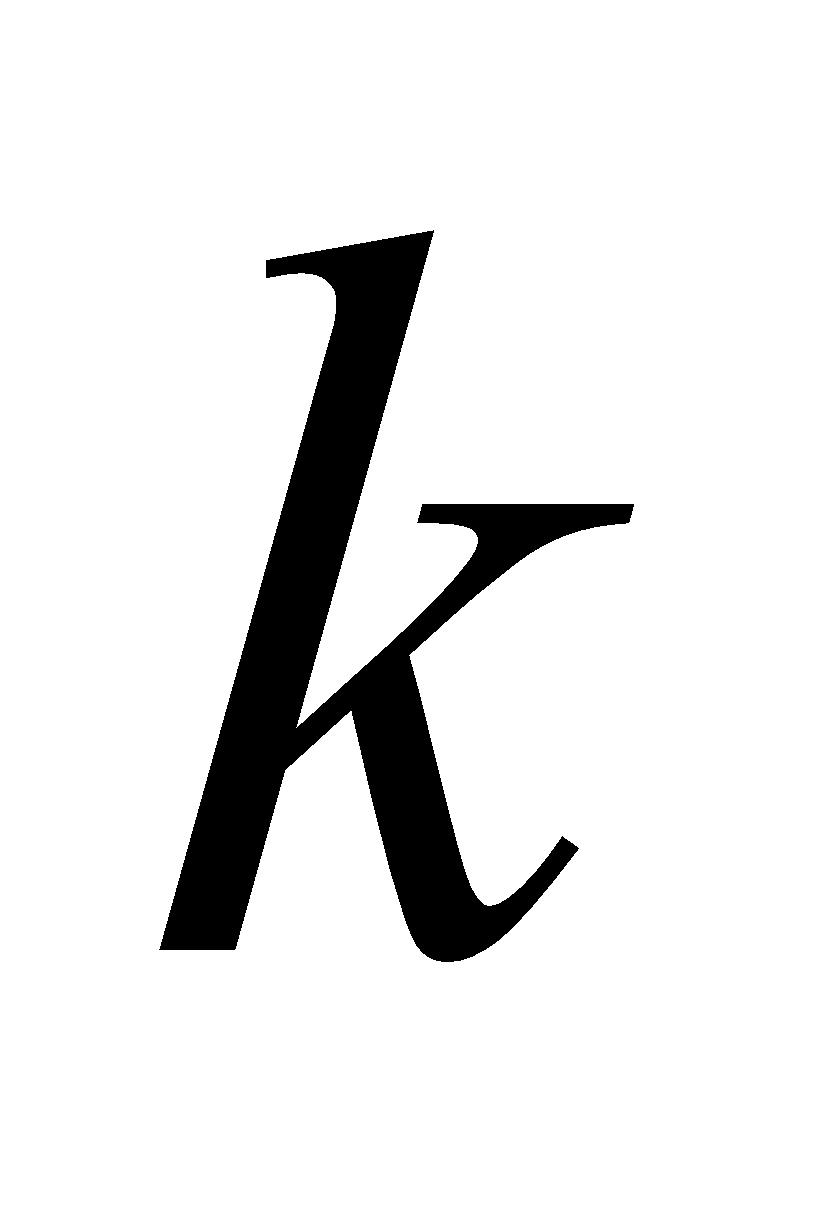
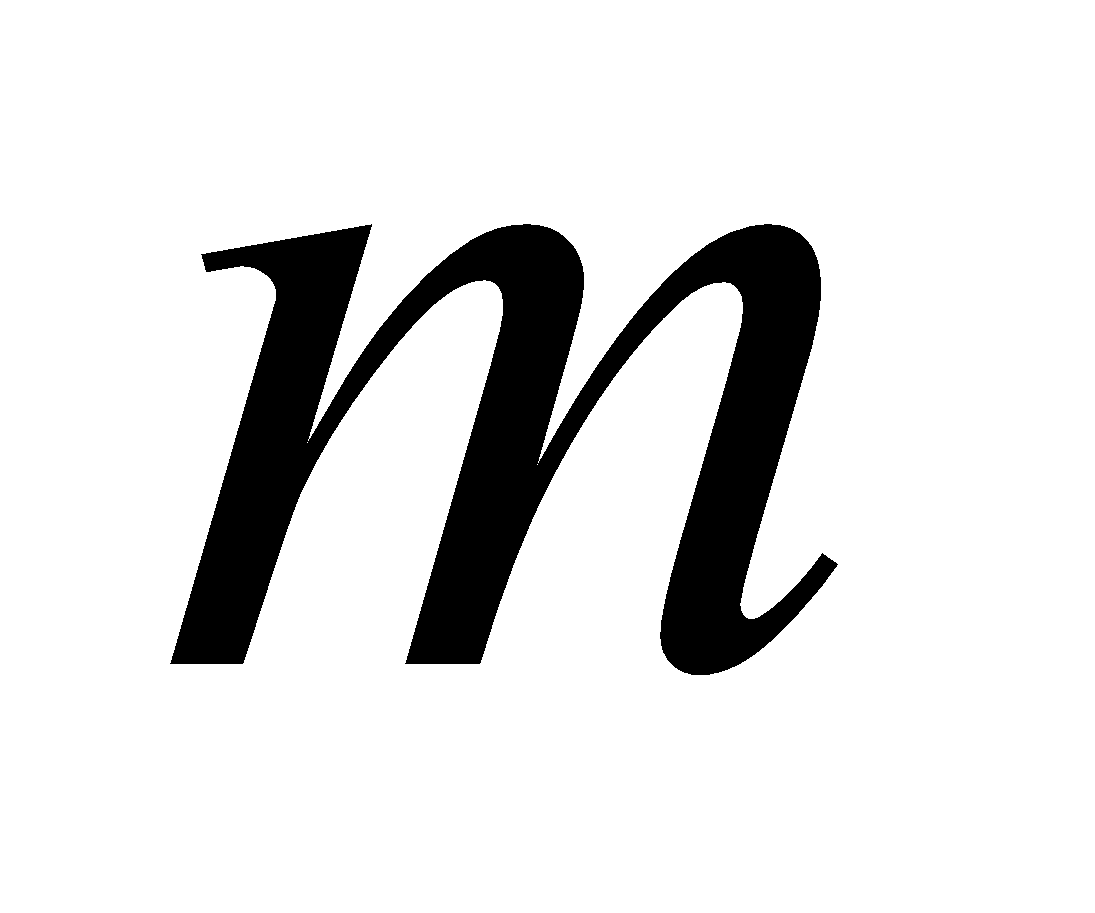
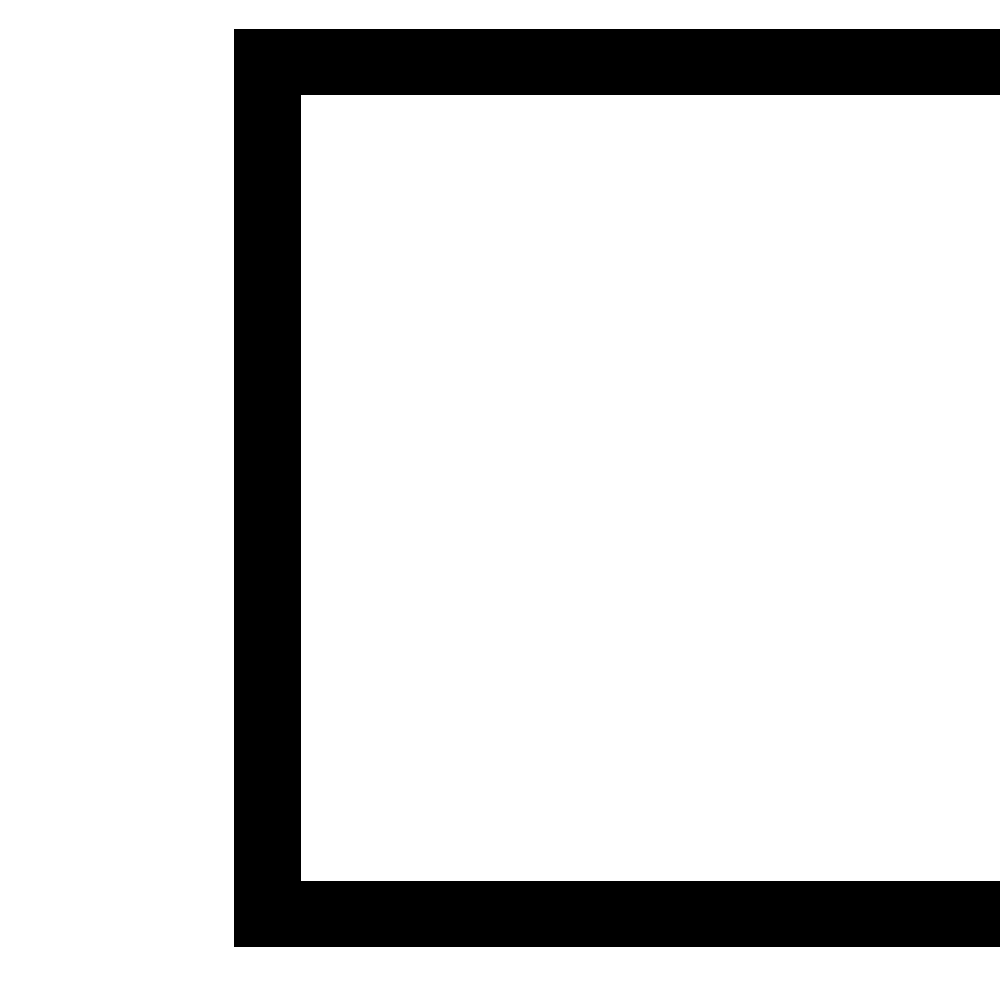
Лемма.  постоянно в таблице  с  состояний.

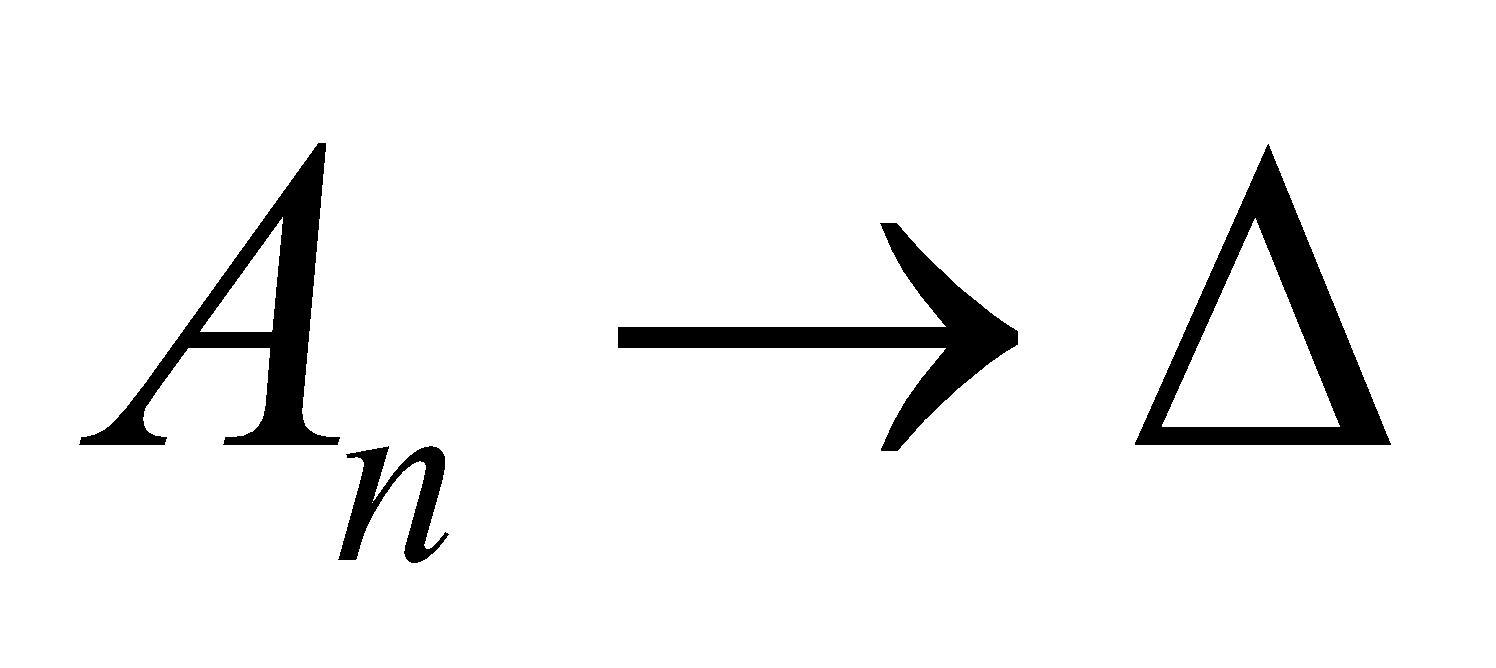
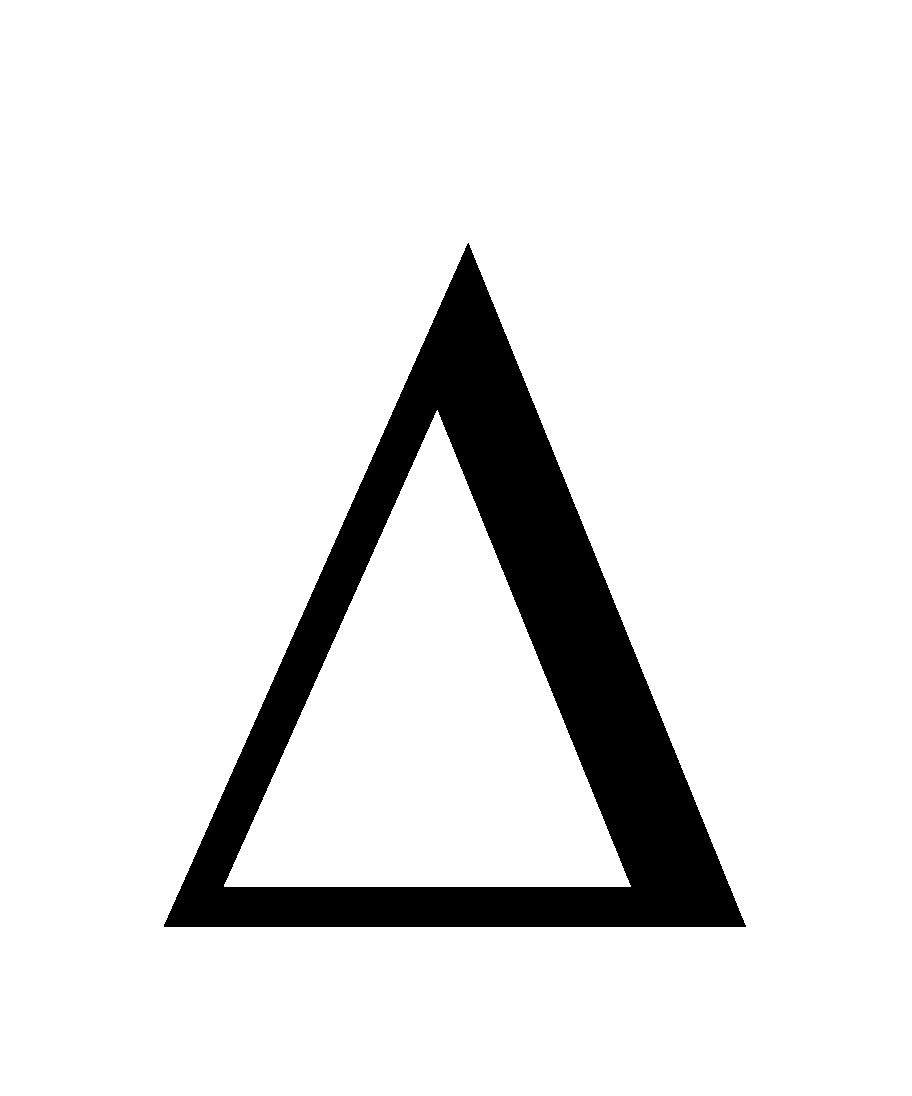
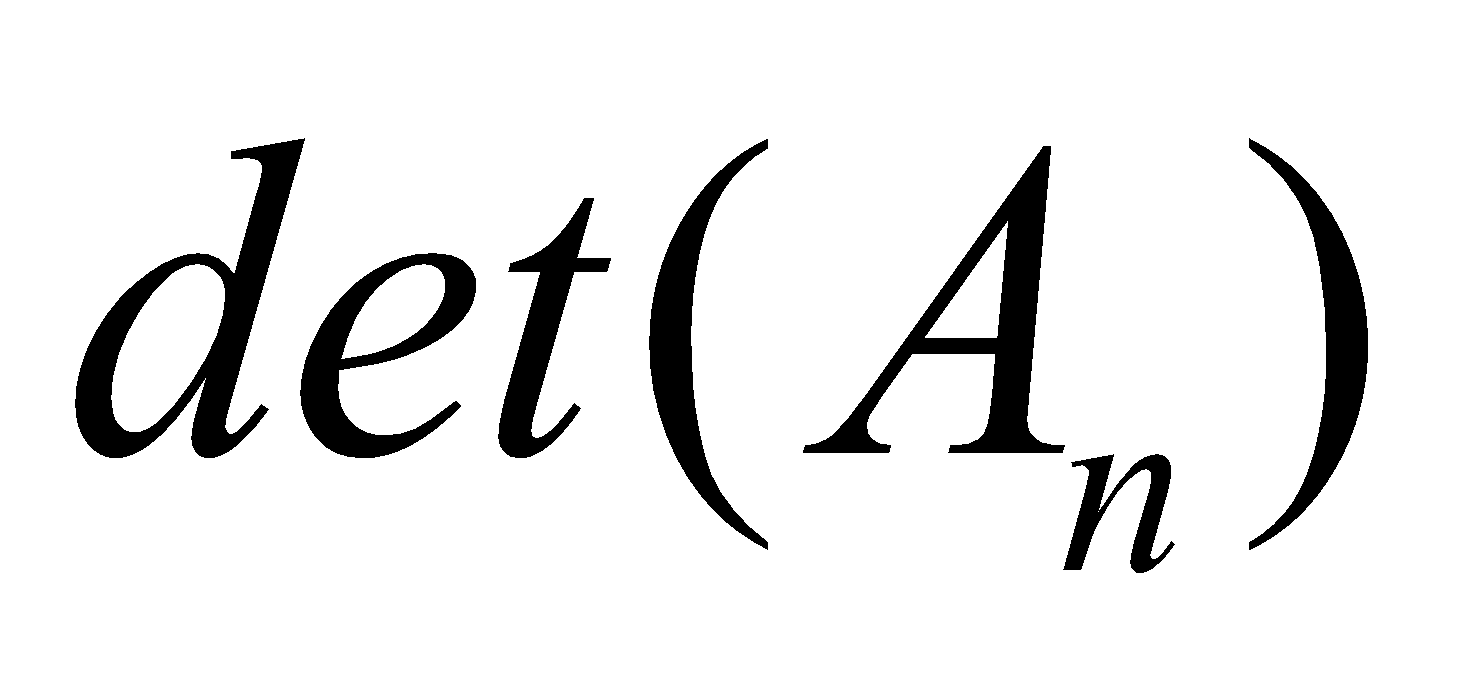
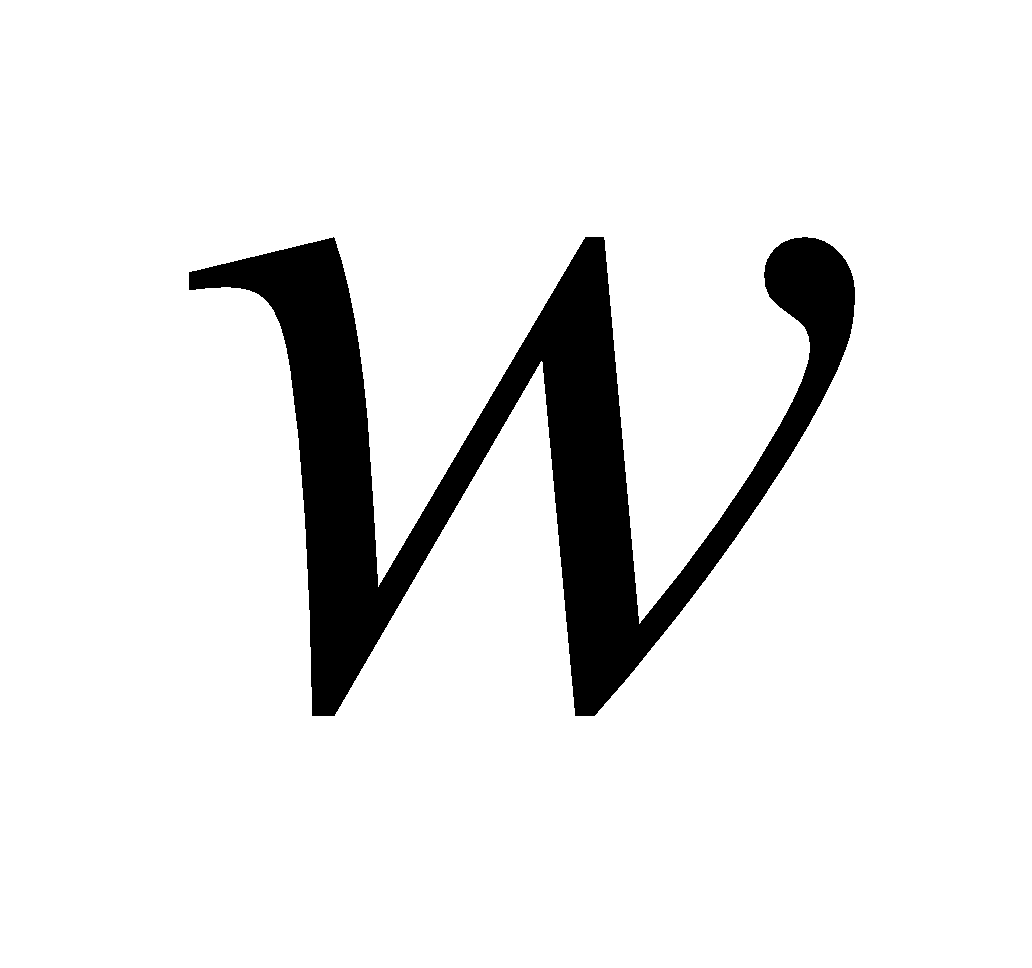
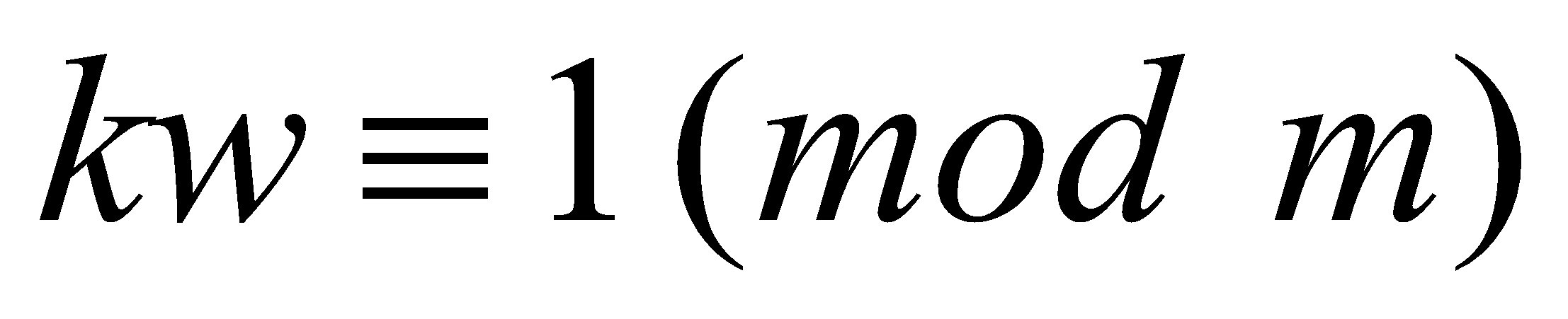
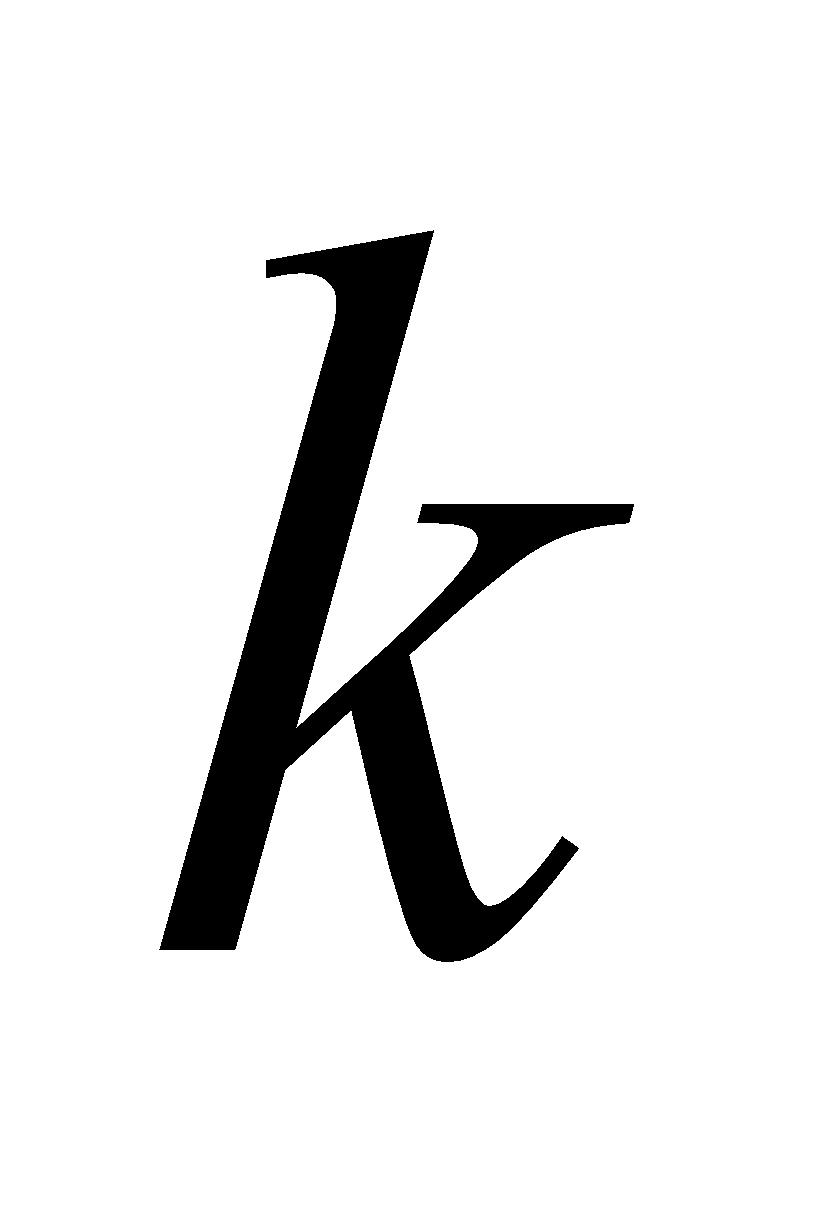
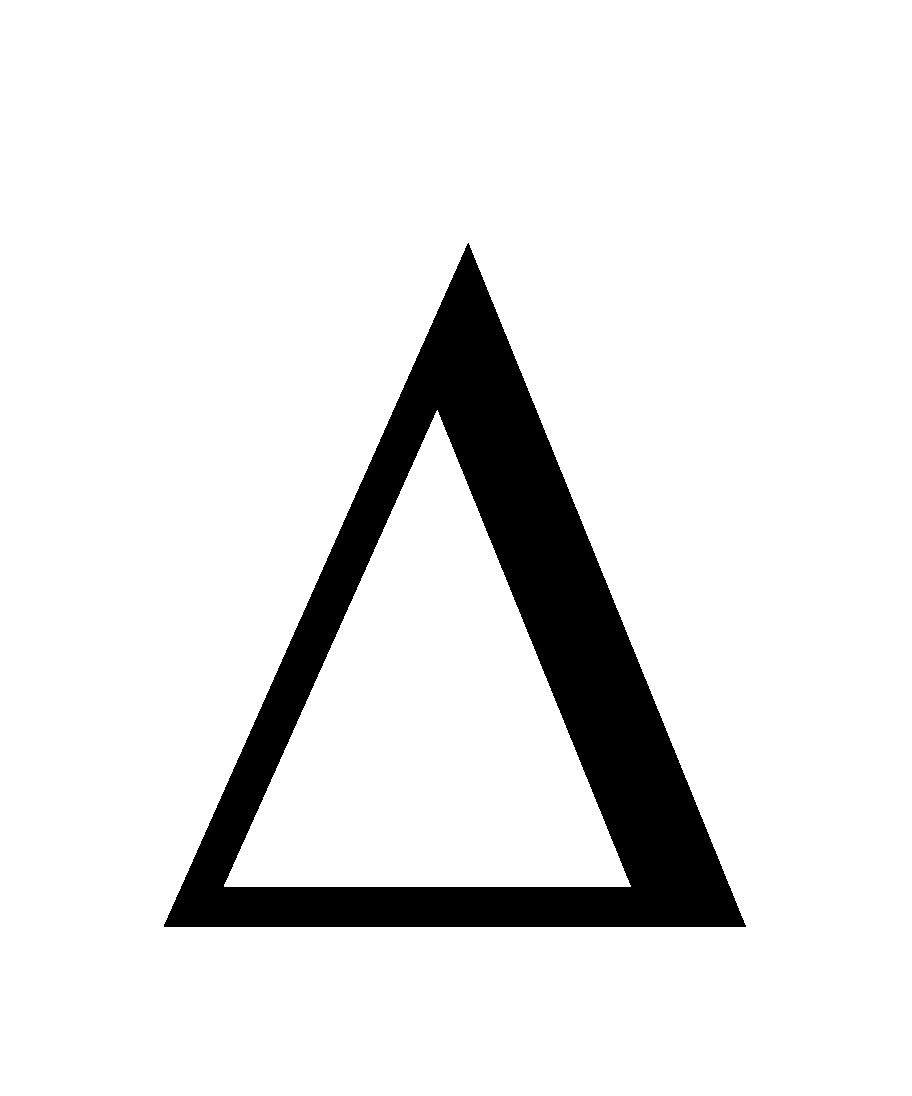
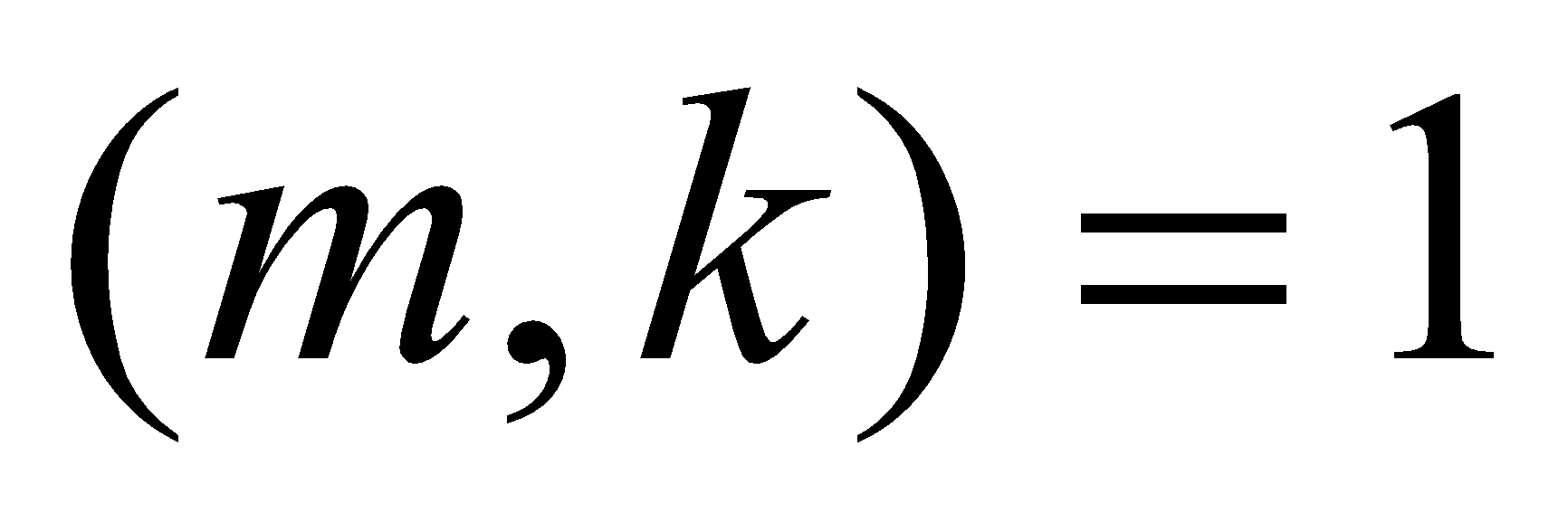
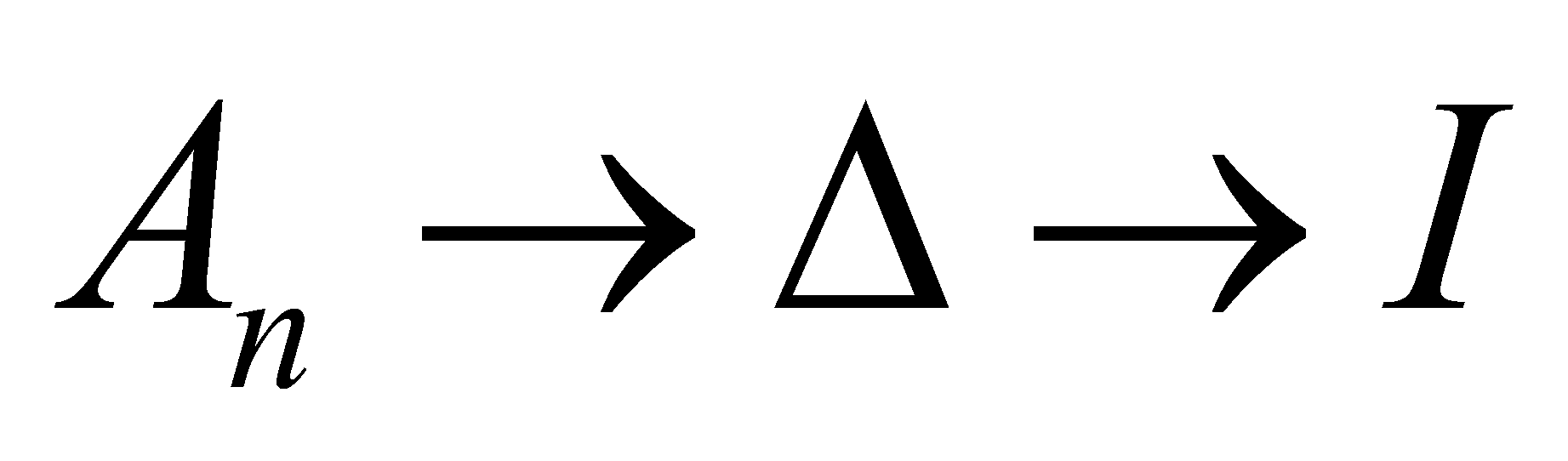
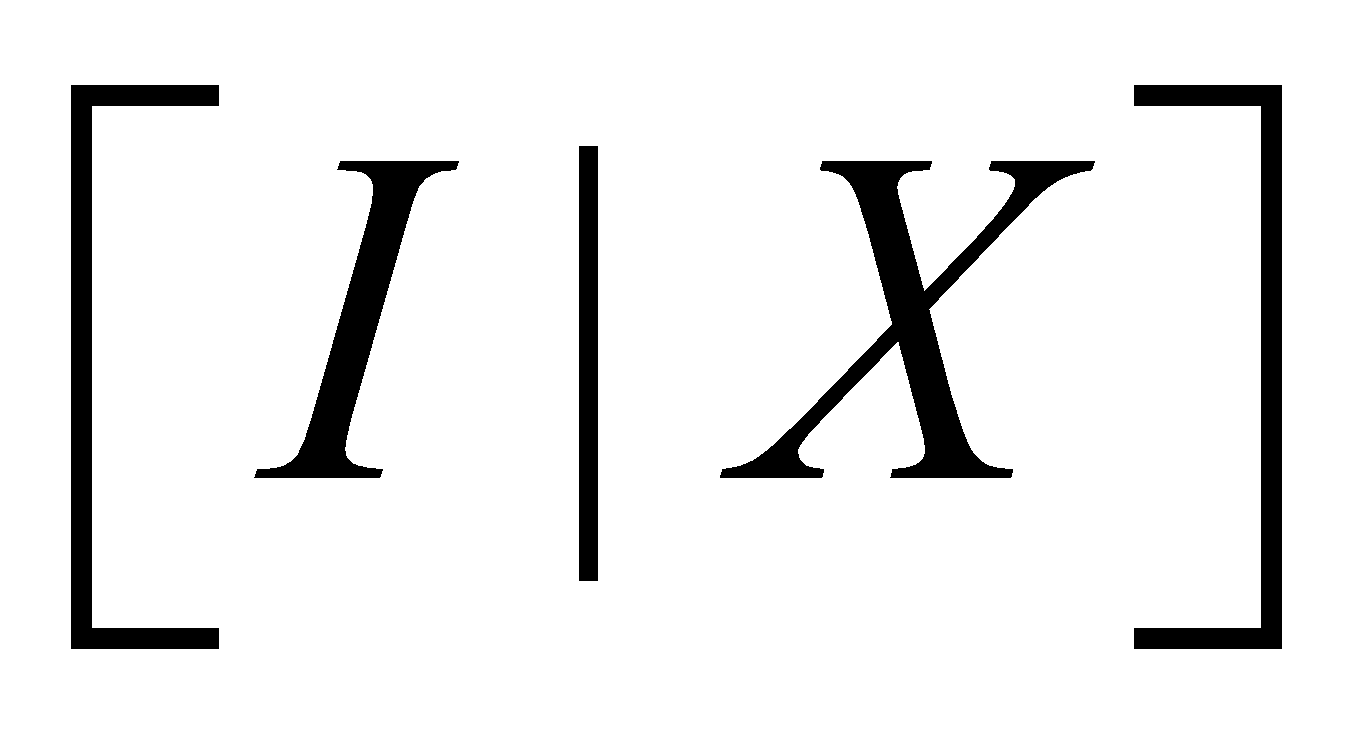
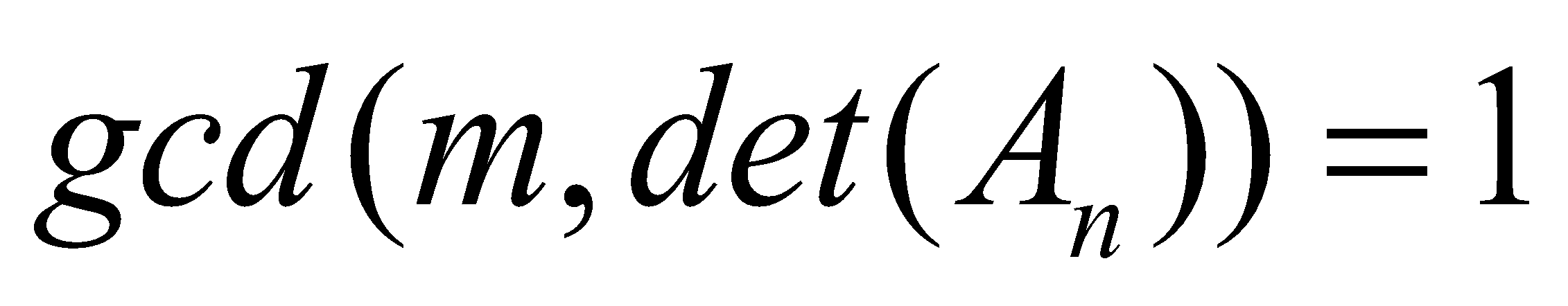
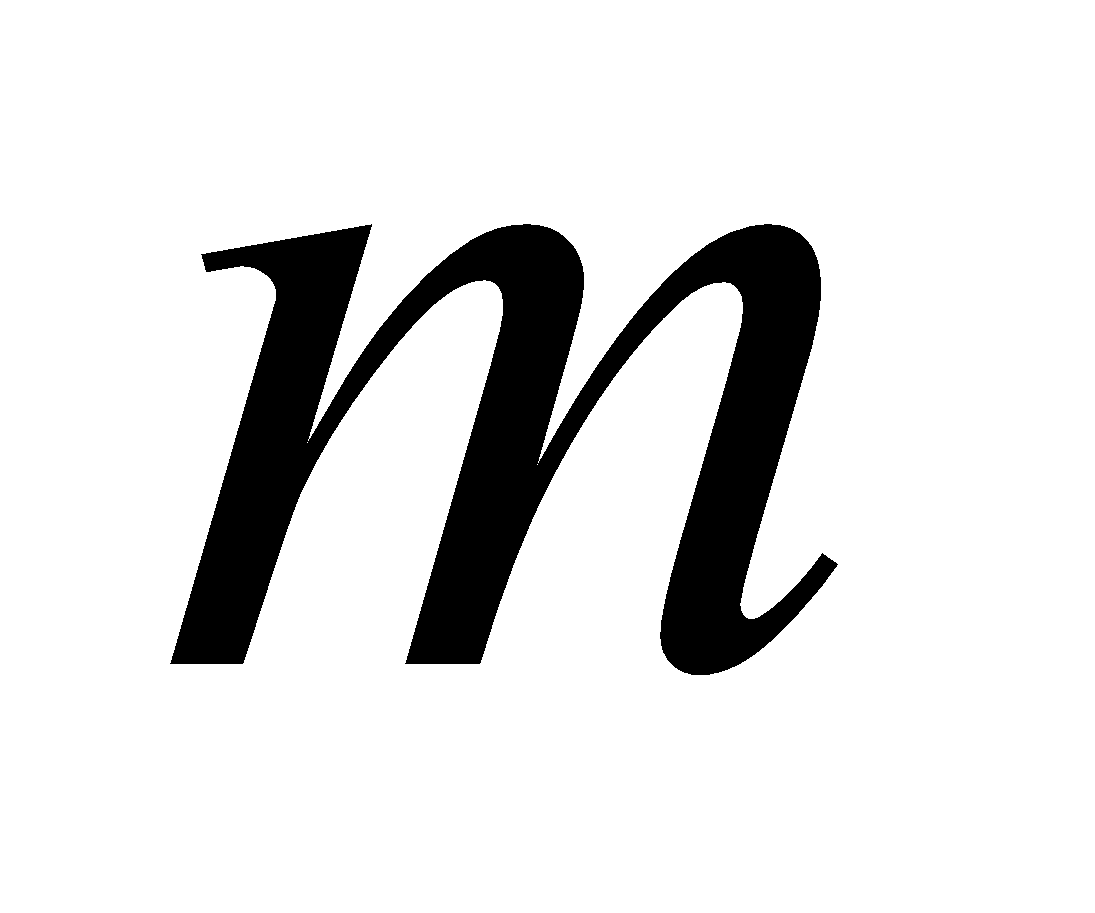
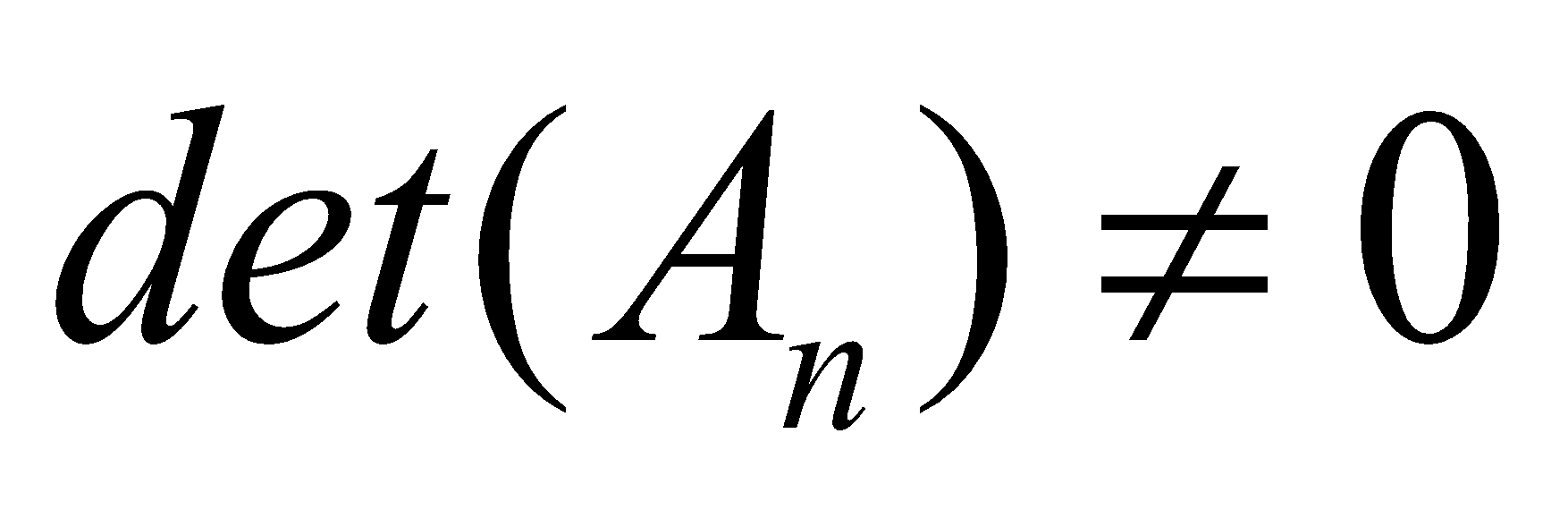
Док-во. Пусть  это набор всех серий нажатий для получения  для некоторых целых  , пусть  это серия нажатий для получения  для некоторой таблицы . Мы получим последовательность  ,где каждый элемент это серия нажатий для получения . Предположим, что существует  , которое является серией нажатий для получения  и . Следовательно, мы получим  являющееся серией нажатий для получения . Таким образом, . Поскольку, то  приводит к противоречию. Таким образом,  является постоянной для таблицы  с  состояниями. 

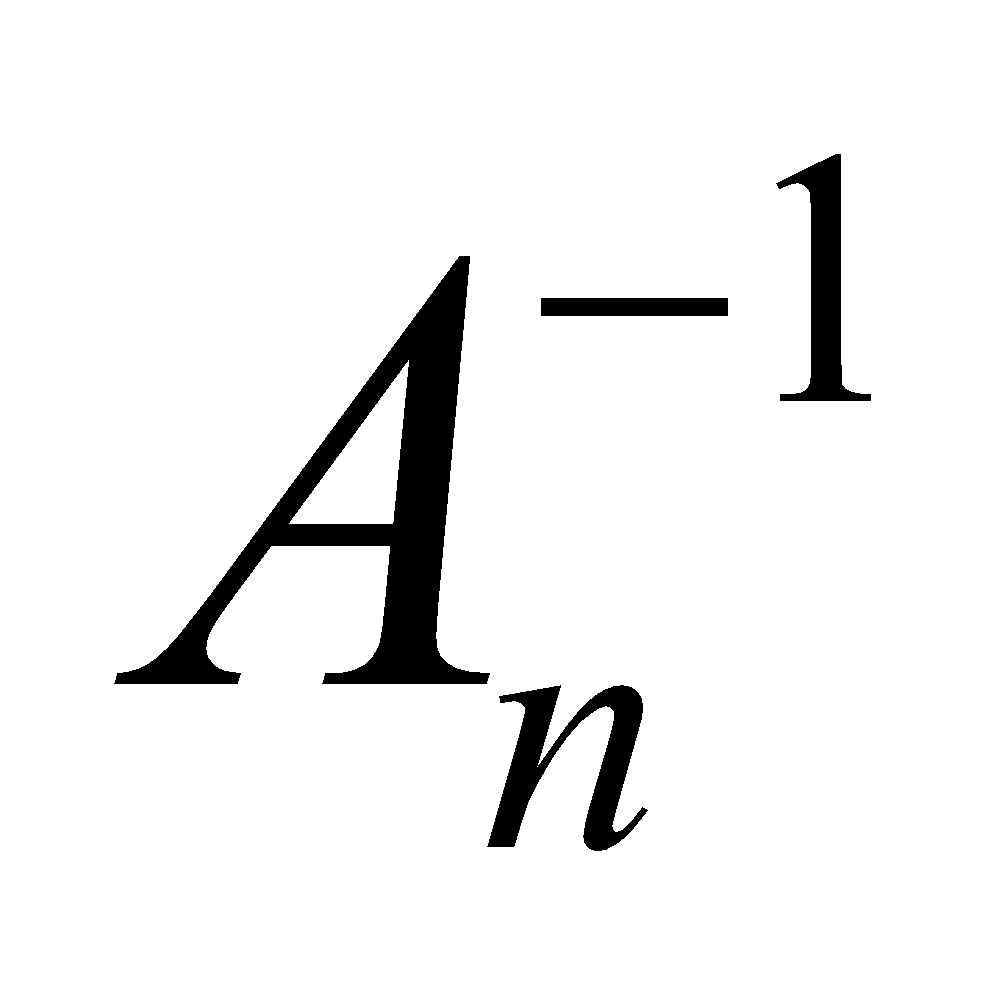
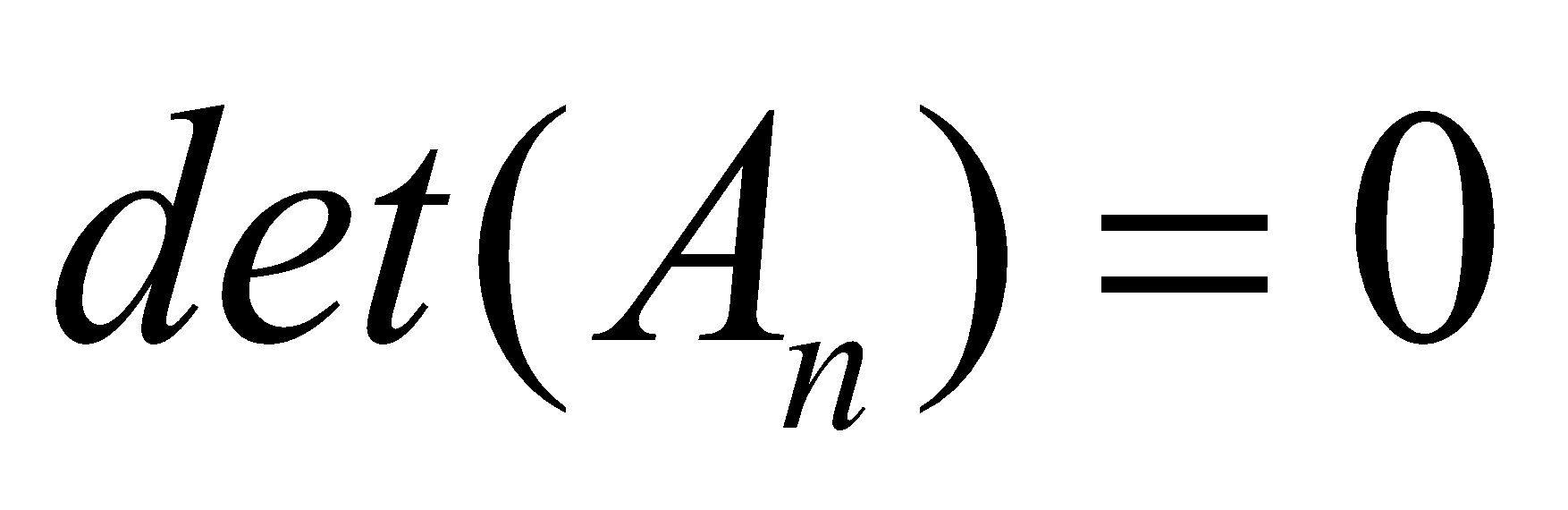
В следующем шаге нашего решения мы будем рассматривать таблицу, для которой мы можем найти решение. Исходя из того, что  и матрица  и мы знаем их значение, мы будем рассматривать  , то есть в расширенной матричной форме. Таким образом, мы можем найти  умножением на . Мы получим . Элементы  могут быть дробью, которую мы не определяли, потому что . И иногда мы не можем найти  поскольку . Таким образом, из  после сведения  где  это ступенчатая форма матрицы. Начиная с этого места, мы заменим  на тождественную матрицу () ,используя лемму. После сведения , мы получим  ,в то время как  будет значением i-й строки .

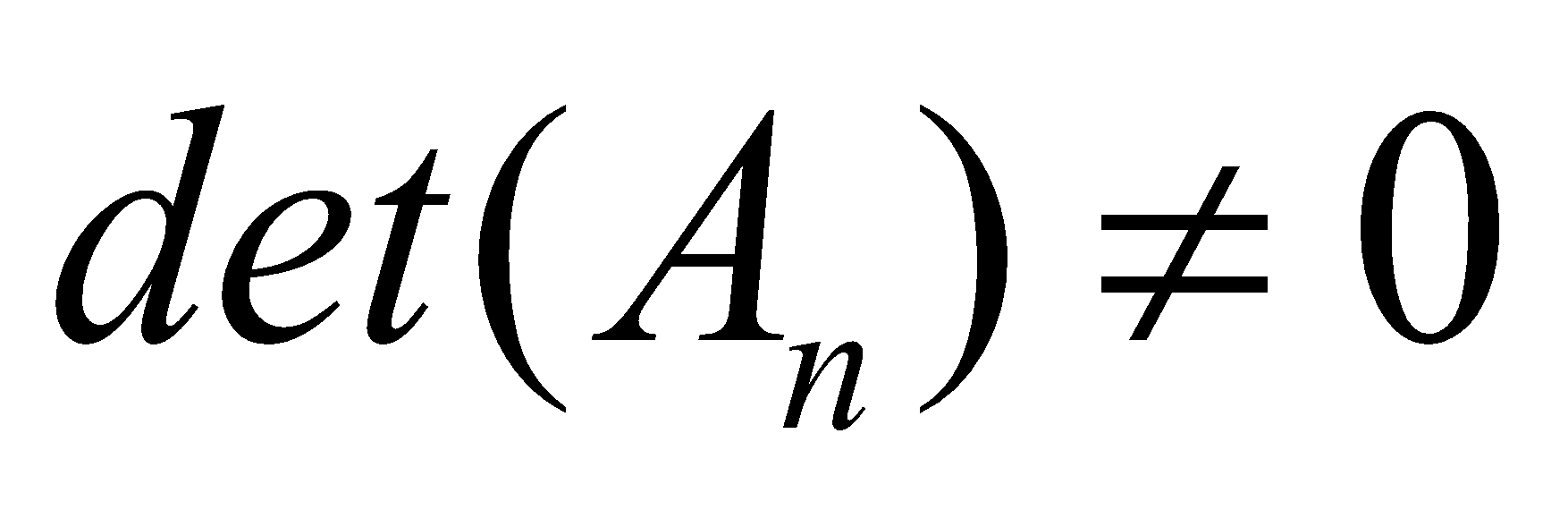
Лемма. Для  , и  где   такое, что .

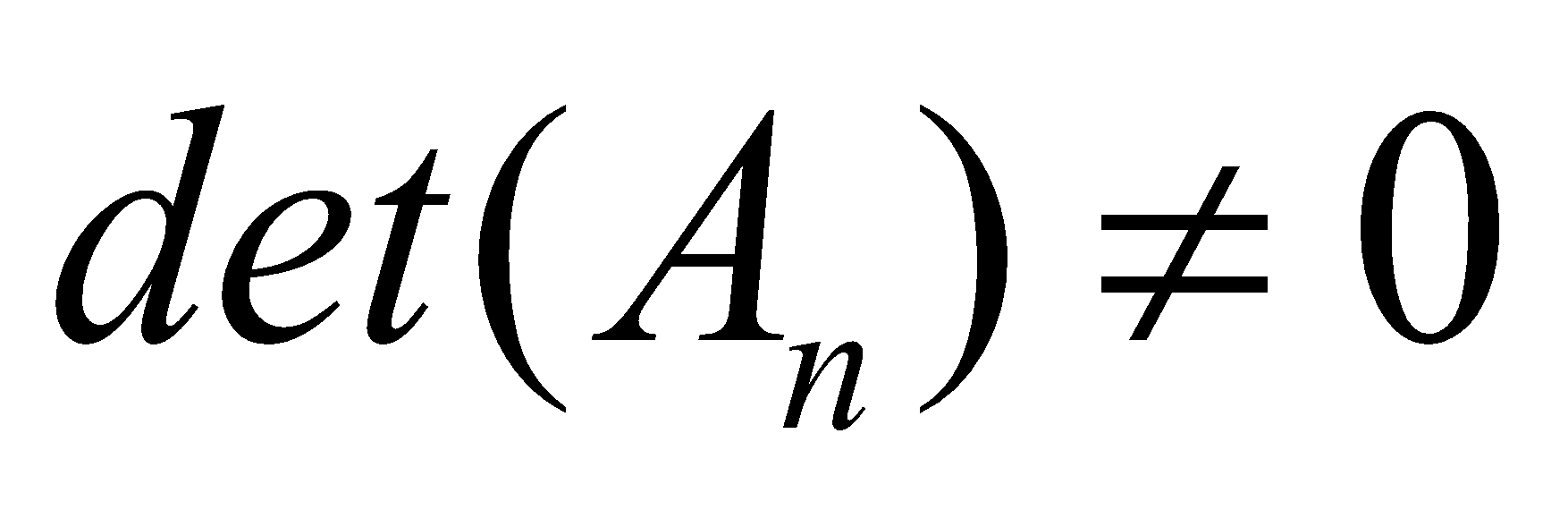
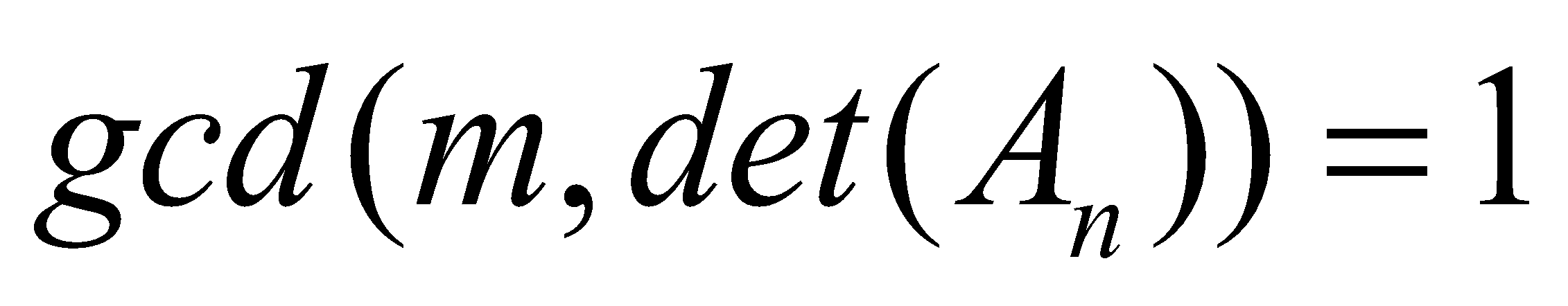
Док-во. Пусть  такие положительные целые числа, что  и .

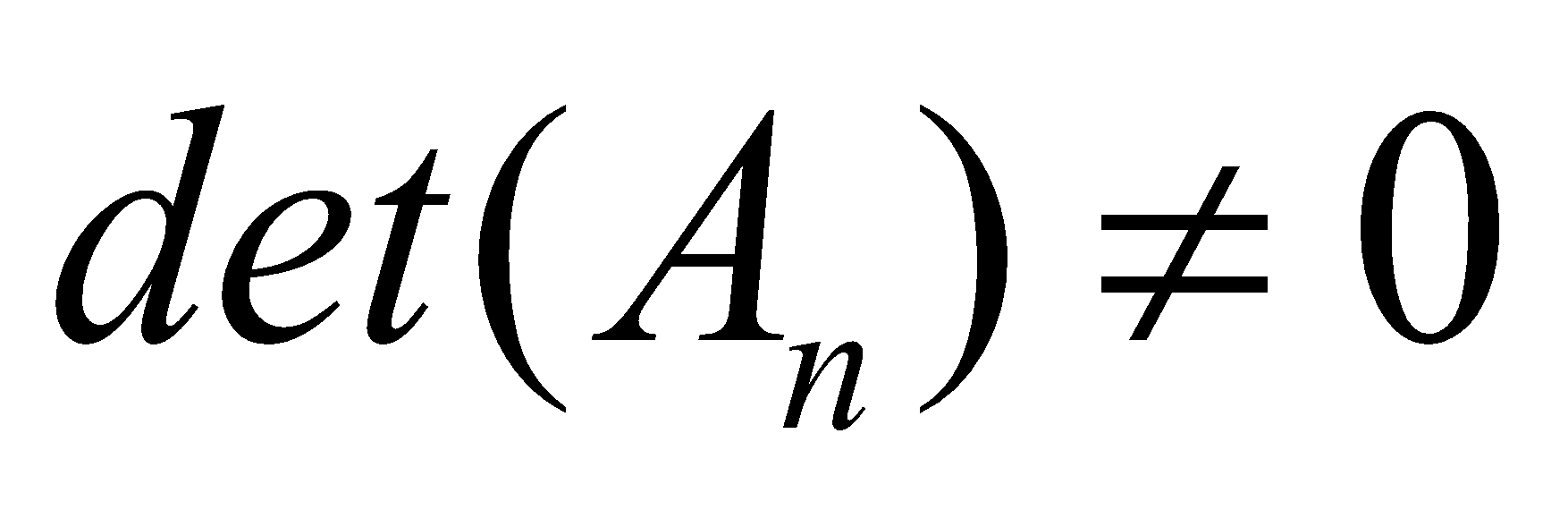
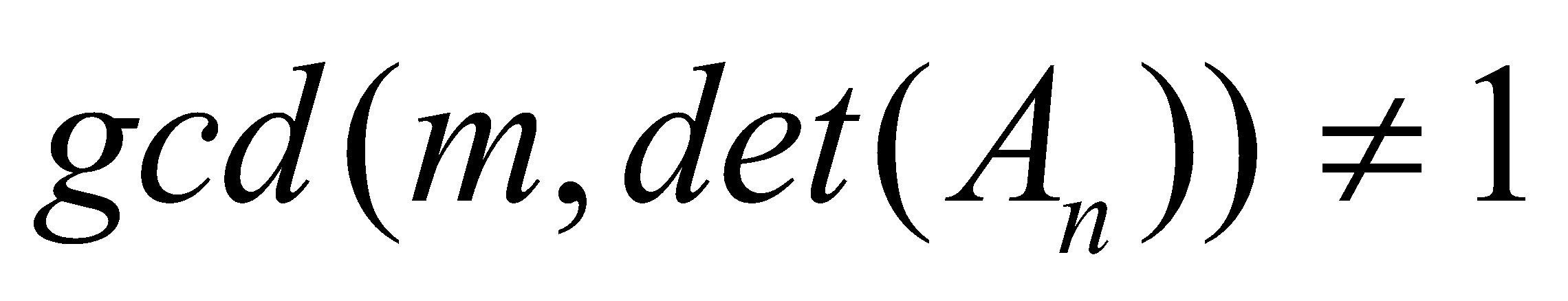
Из тождества Безу мы знаем, что  , где ,  что . Поэтому, . Мы назовем мультипликативно обратным для  по модулю .

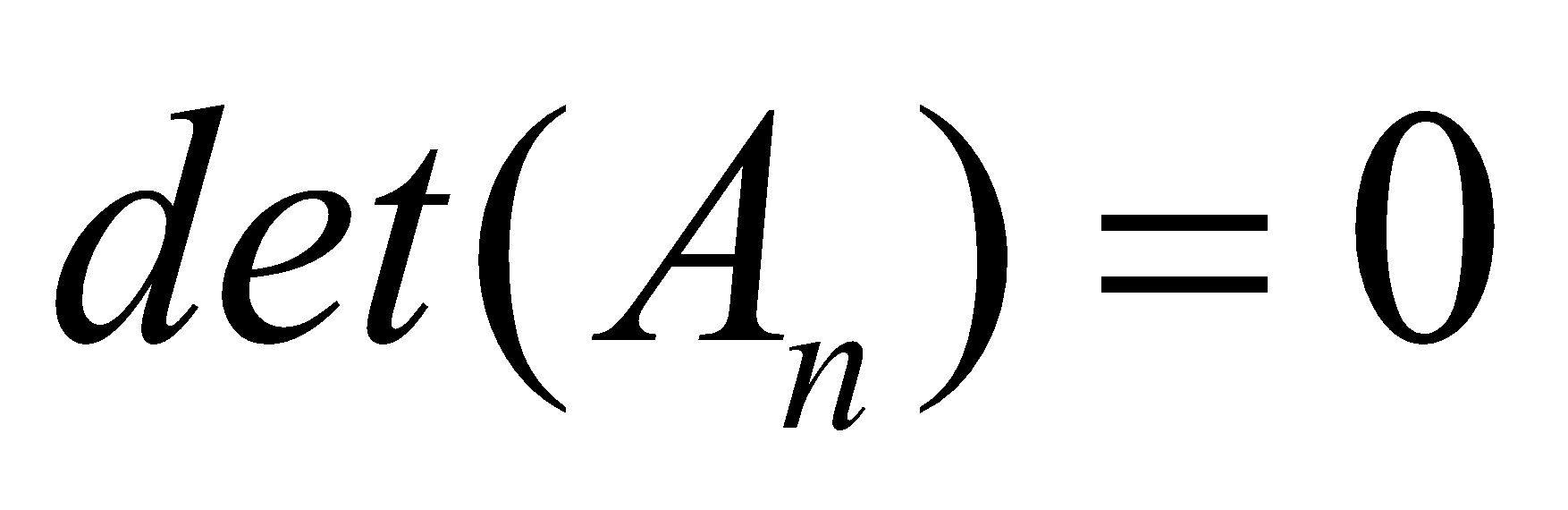
Благодаря сведению , мы знаем, что произведения значений  на диагонали равно . Согласно лемме, мы можем найти  , при котором  для каждого  ,лежащего на диагонали  , причем . Таким образом, мы можем преобразовать  и получим . Таким образом, мы всегда можем найти ответ для начальной таблицы, если  , где  состояние начальной таблицы и .

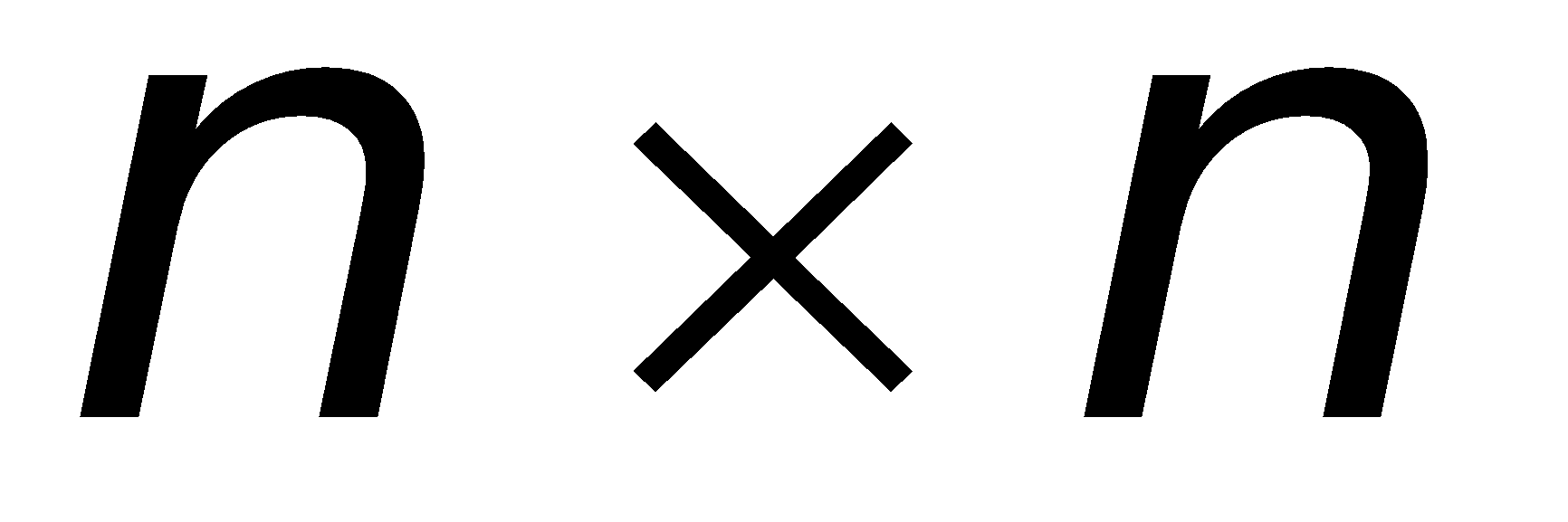
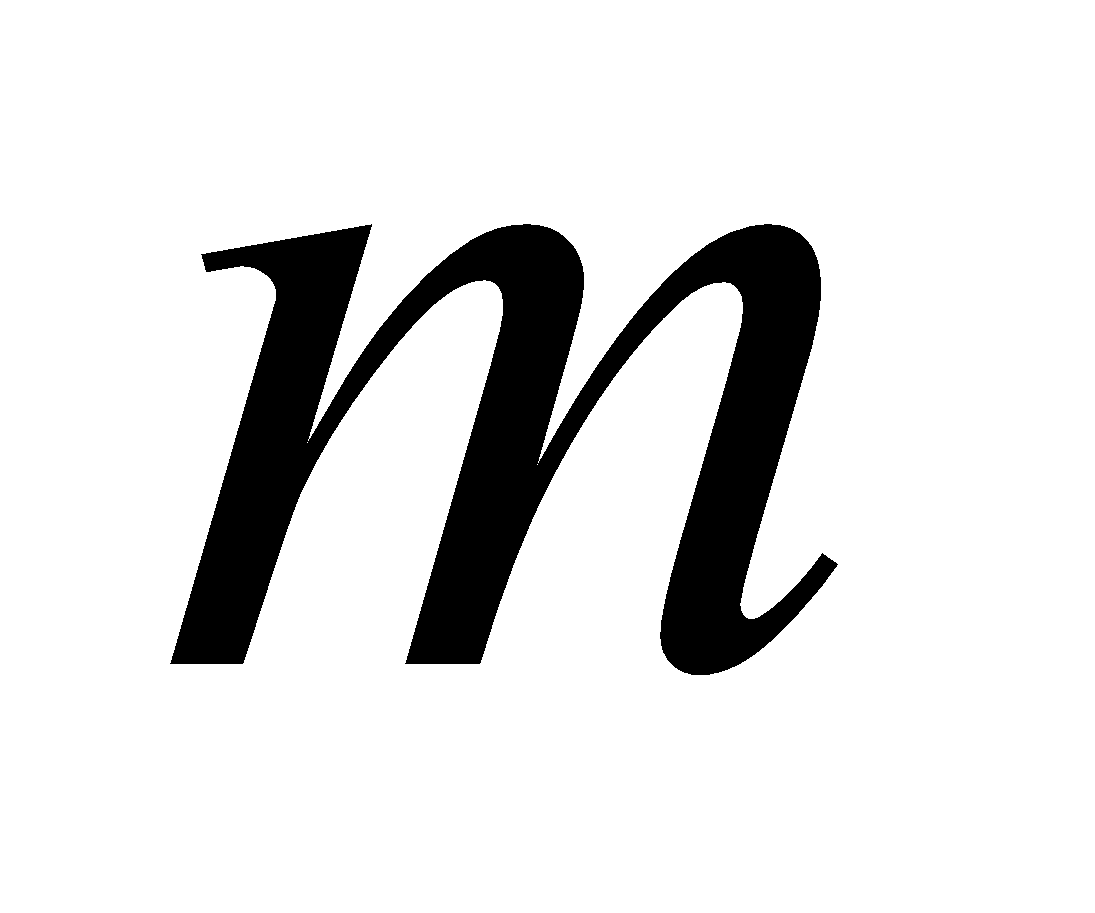
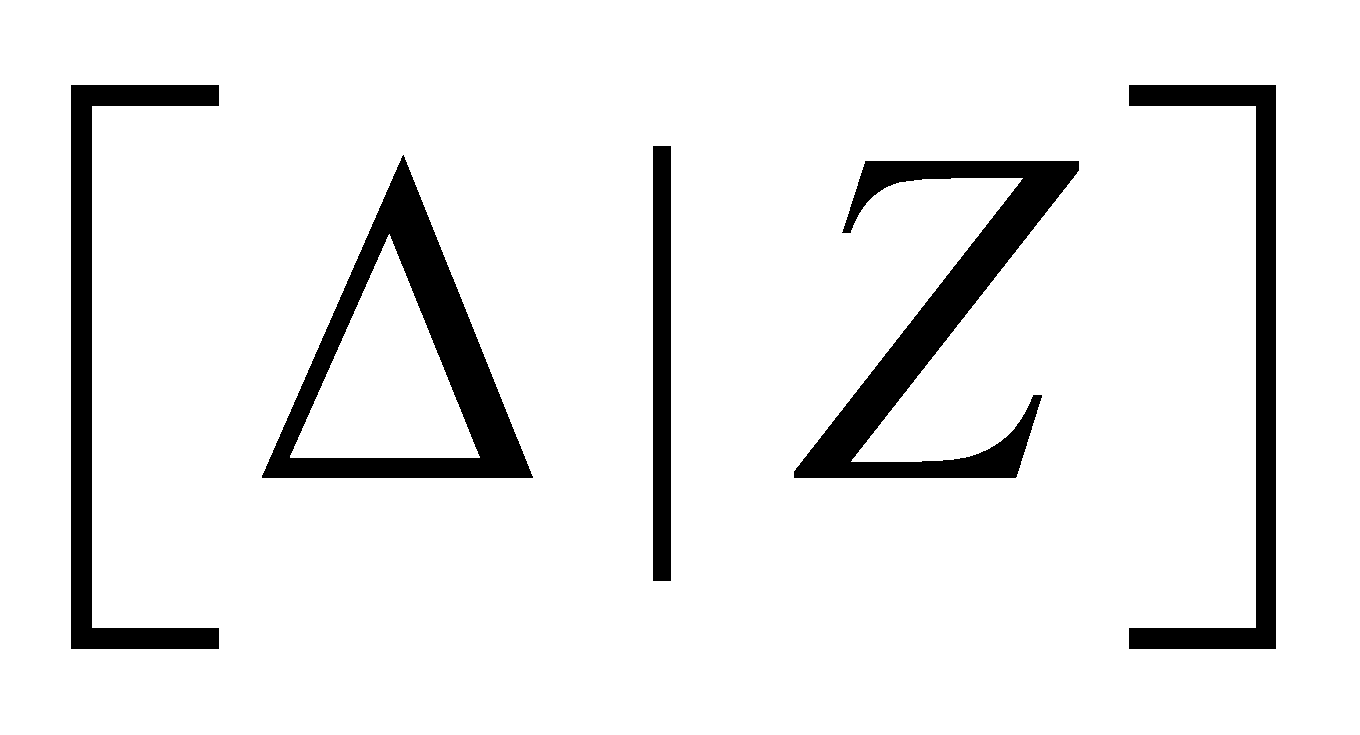
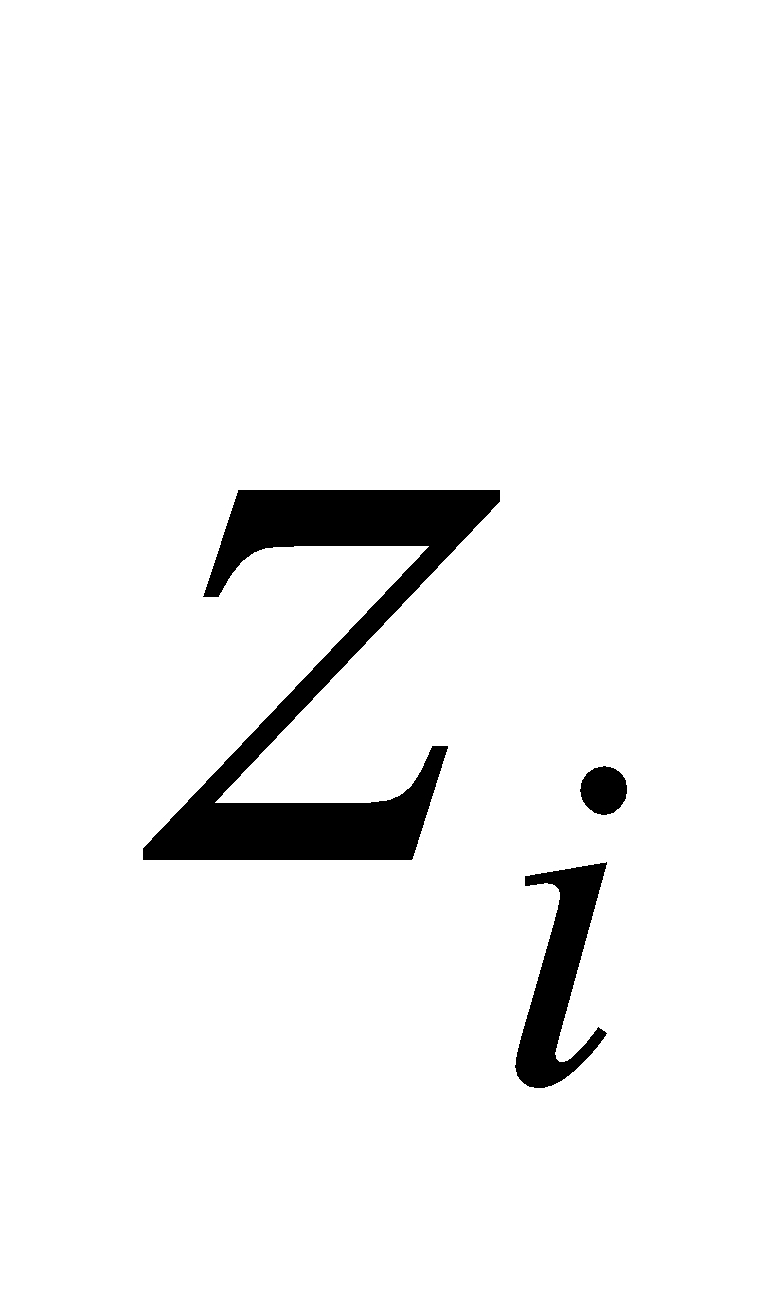
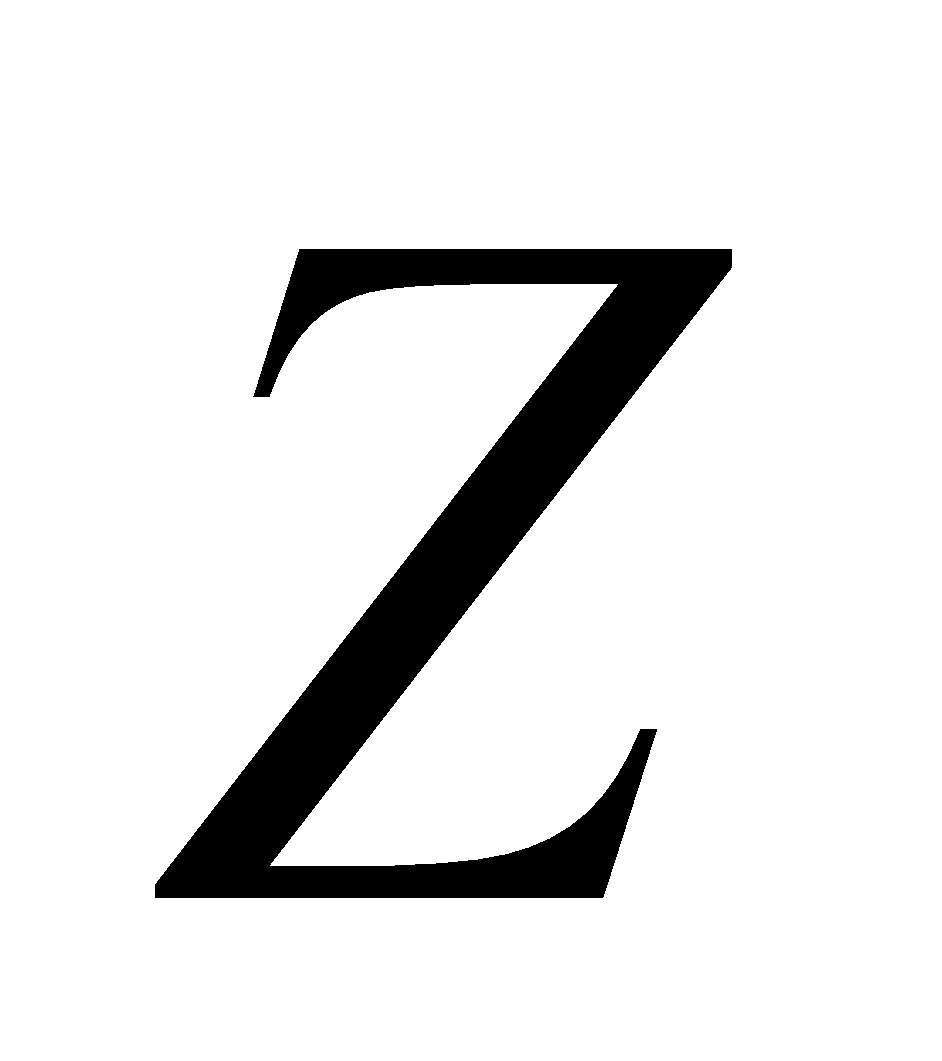
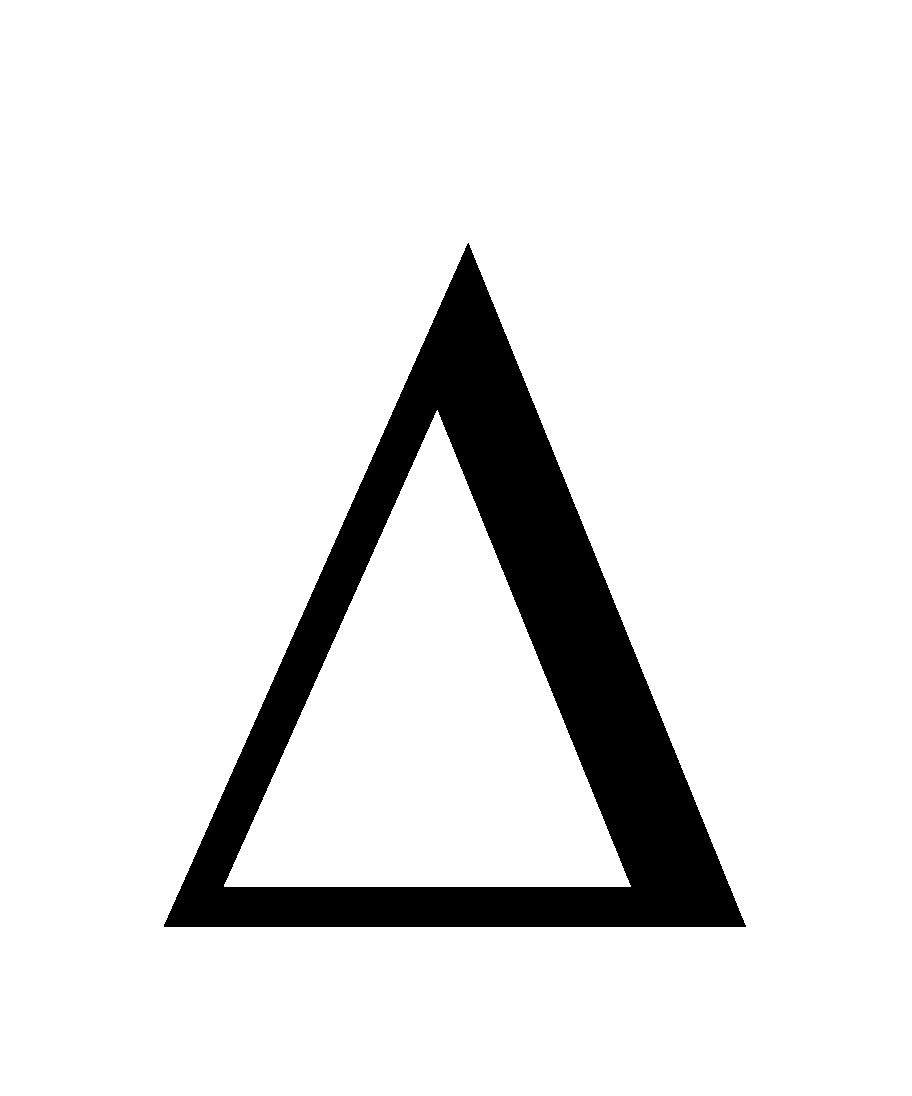
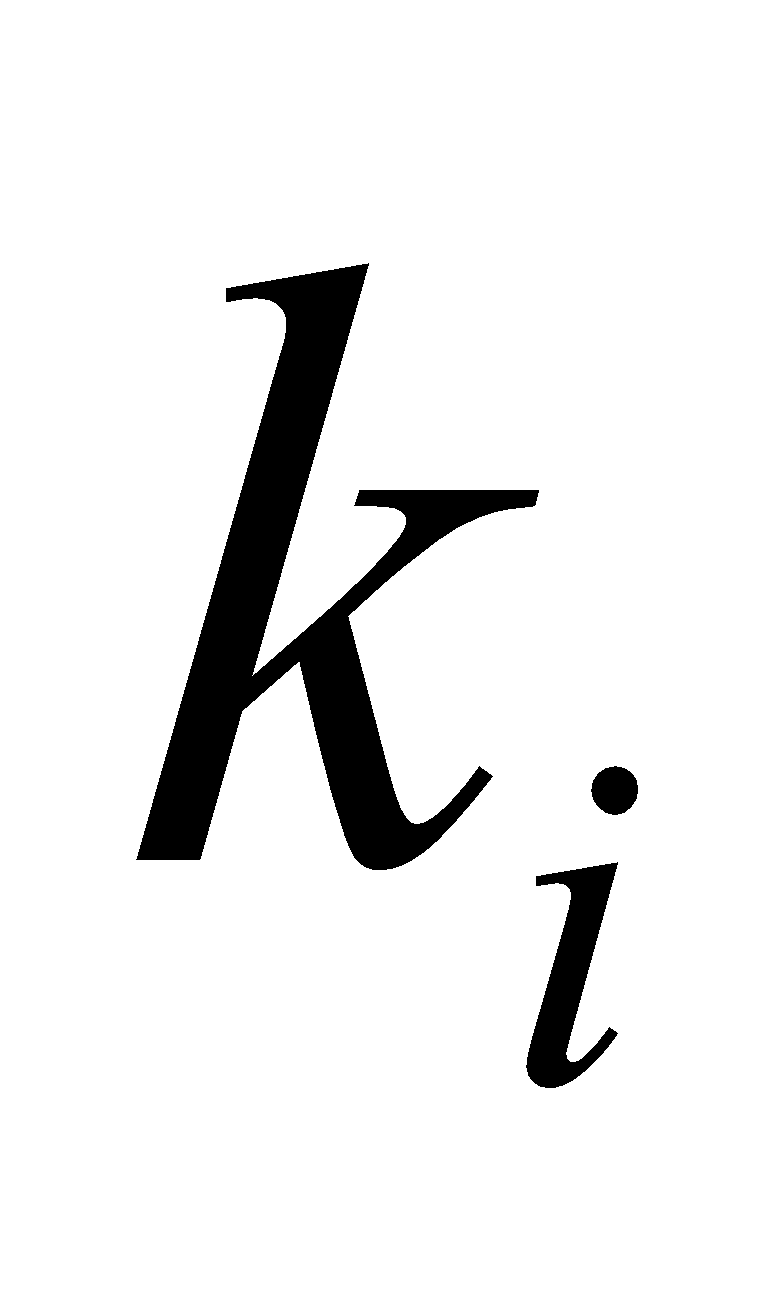
Исходя из сказанного выше, это единственный случай, когда начальная таблица имеет решение. Мы знаем, что существуют таблицы, для которых мы не можем найти , то есть когда . Таким образом нужно рассматривать следующие случаи:

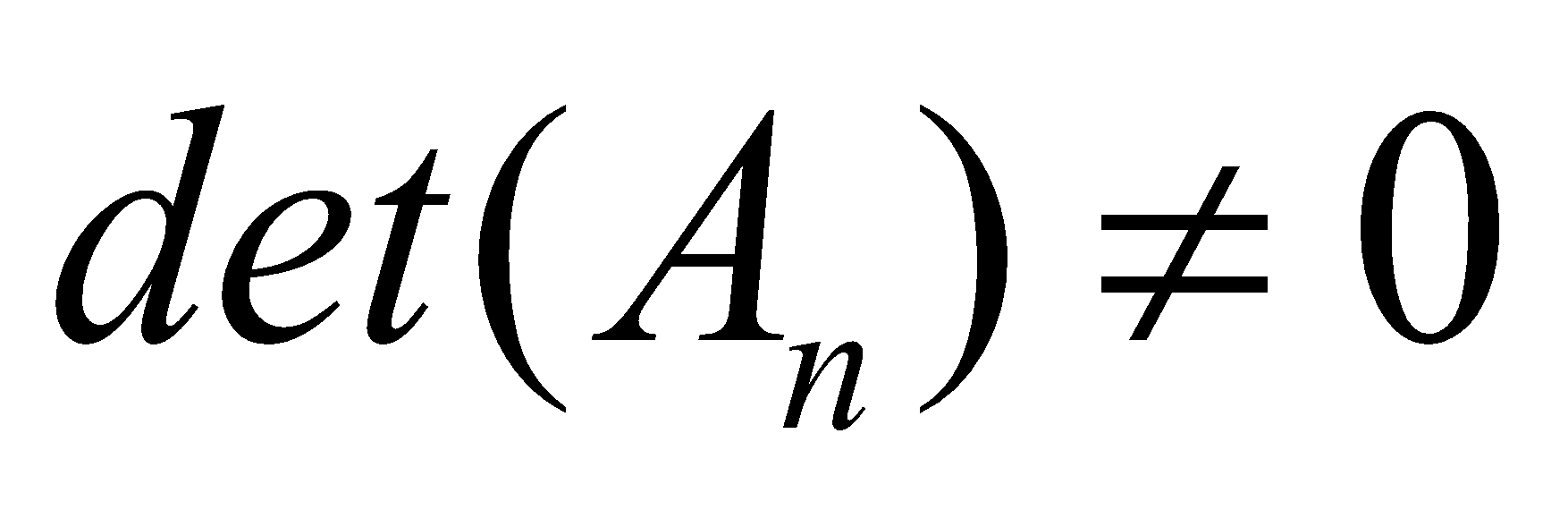
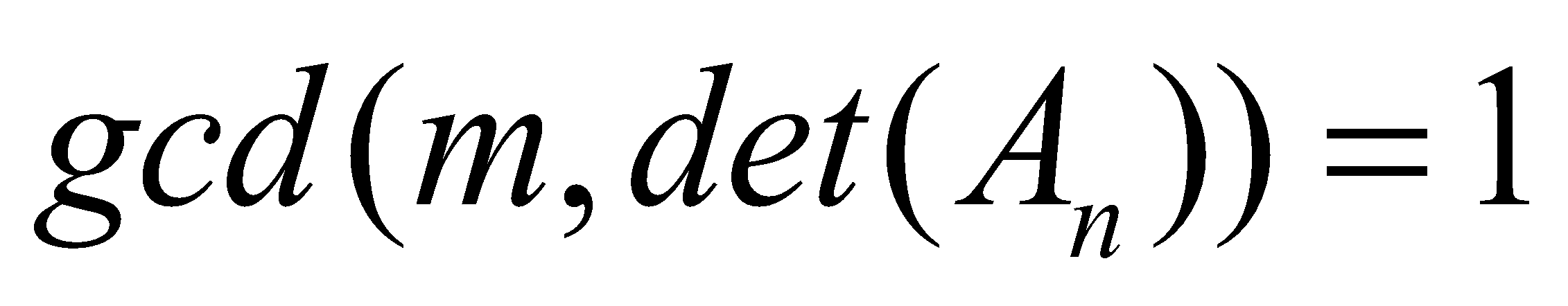
1. 

1.1  и 

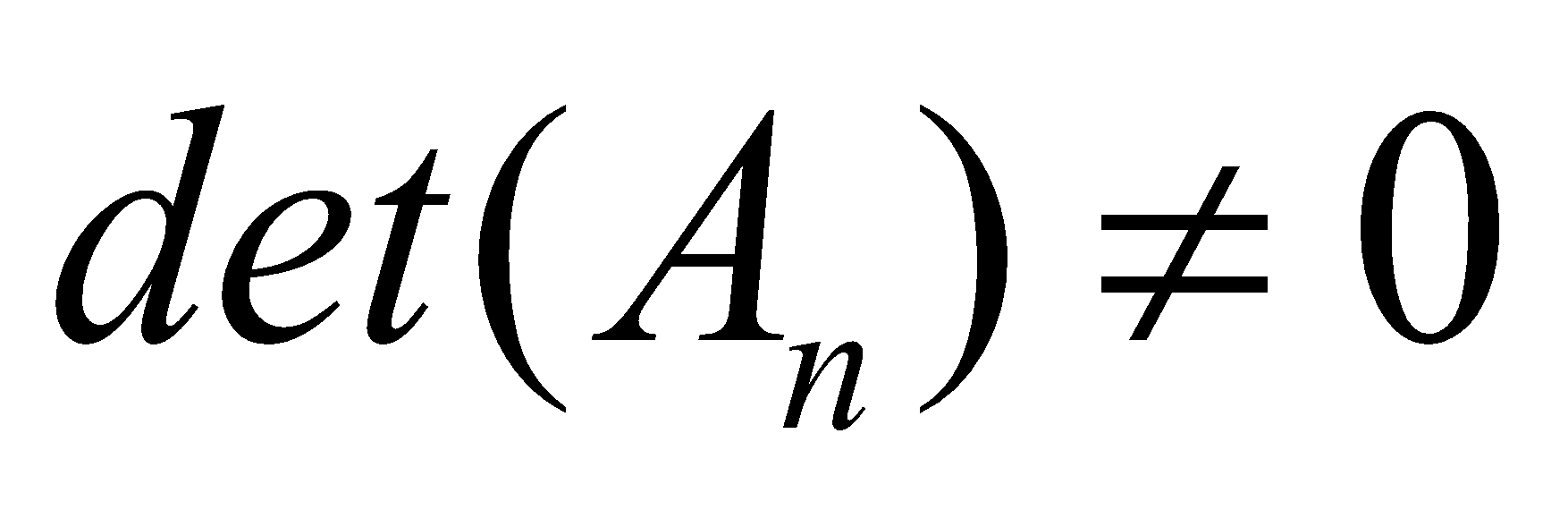
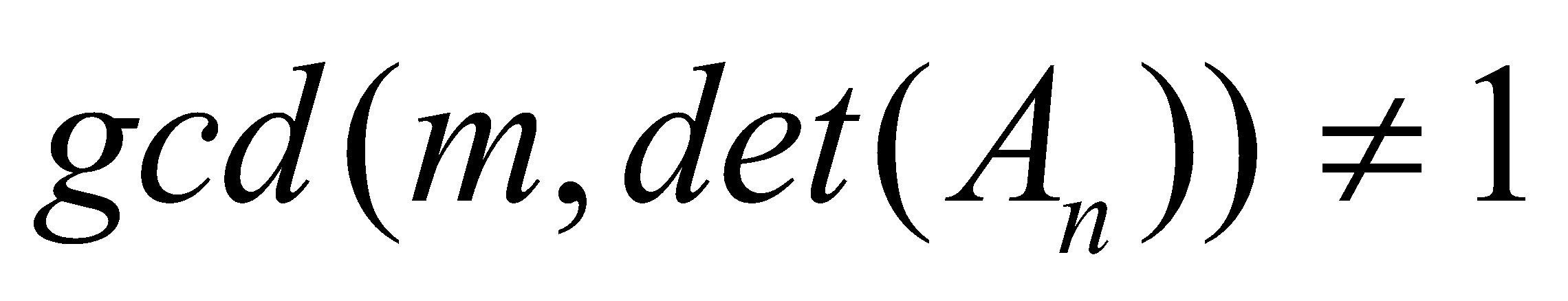
1.2  и 

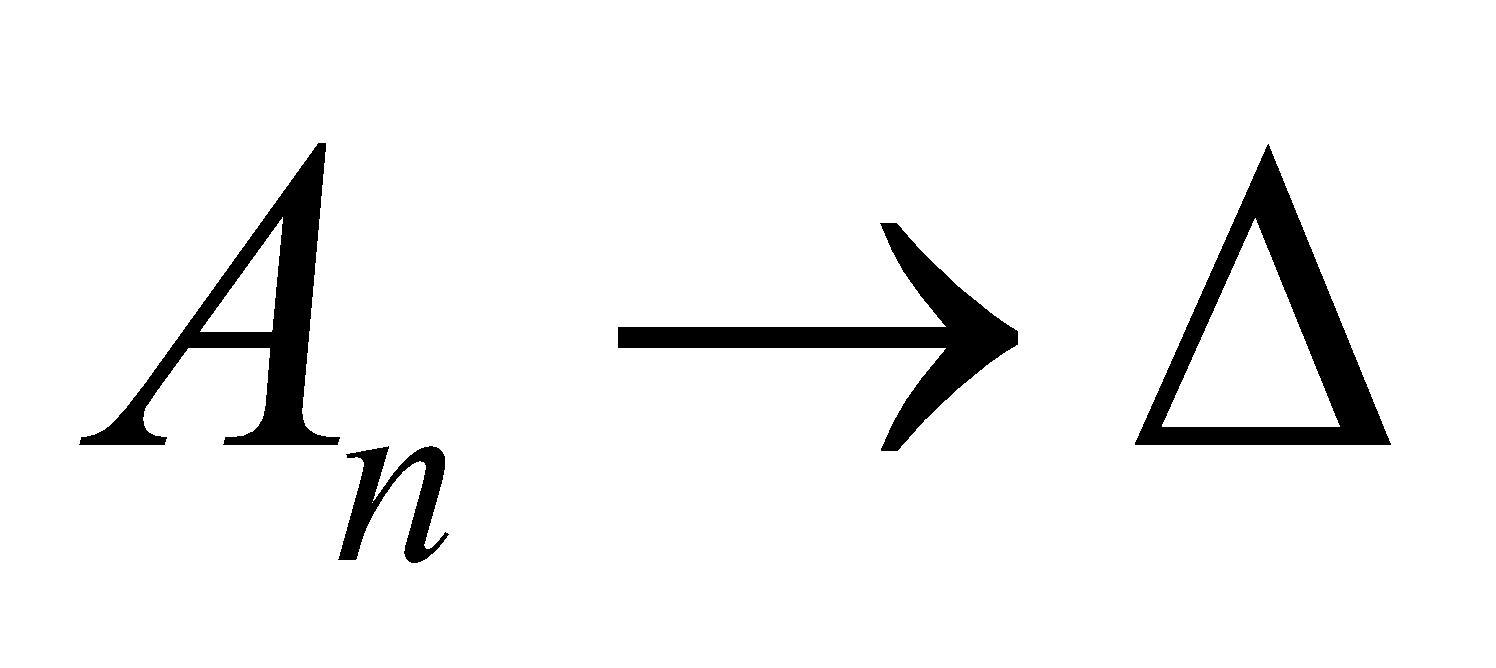
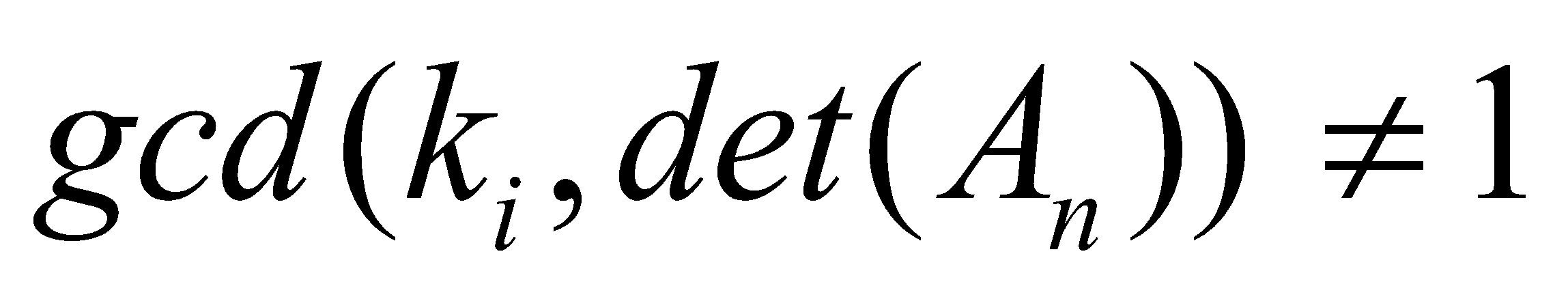
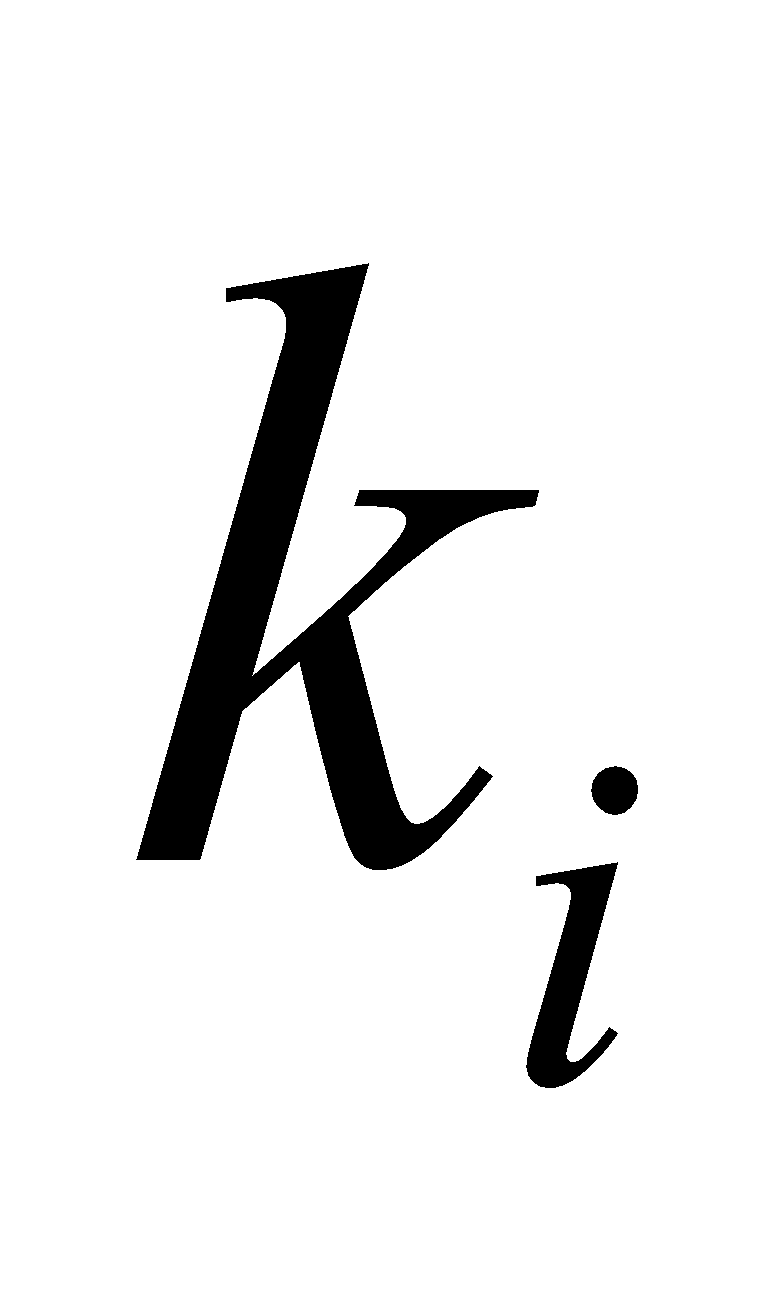
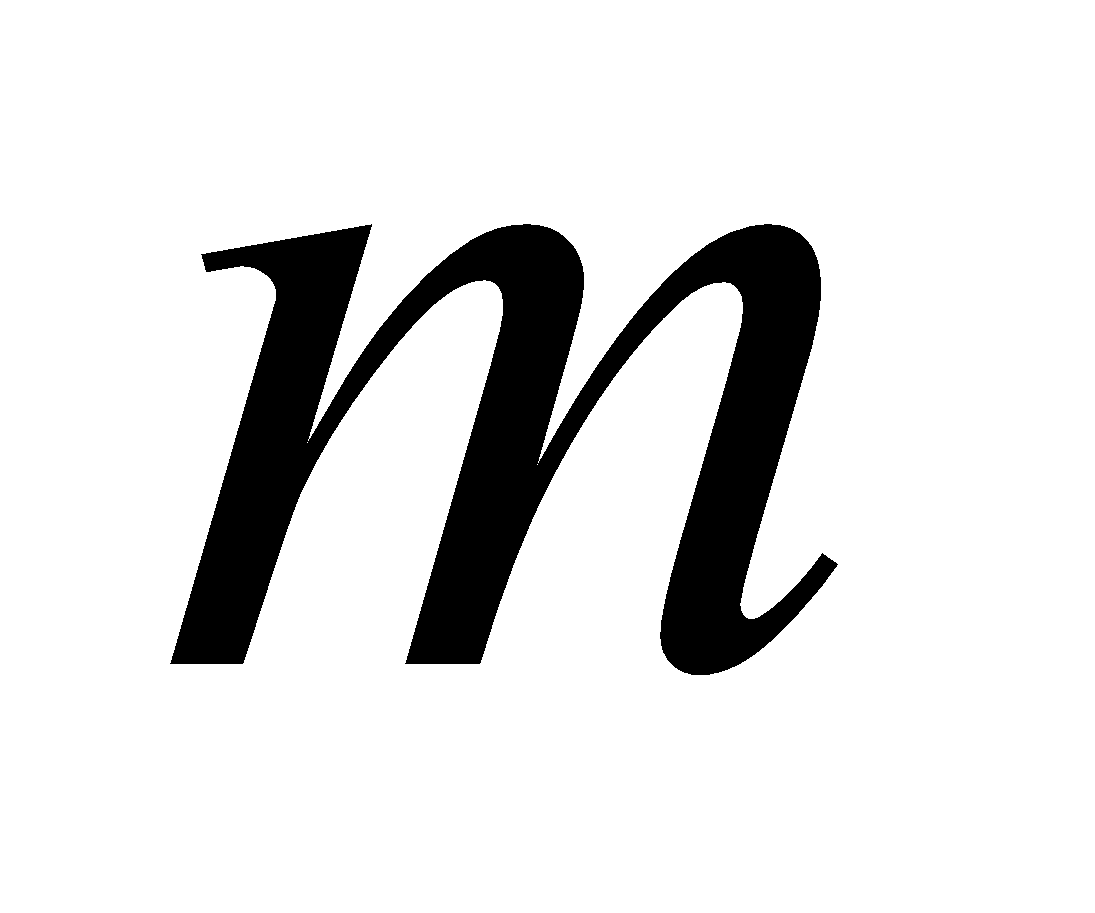
2. 

Для всех случаем мы будем рассматривать таблицу  с  состояний, начиная с  где  значение i-й строки  а число, которое лежит на диагонали в i-м ряду обозначим как .

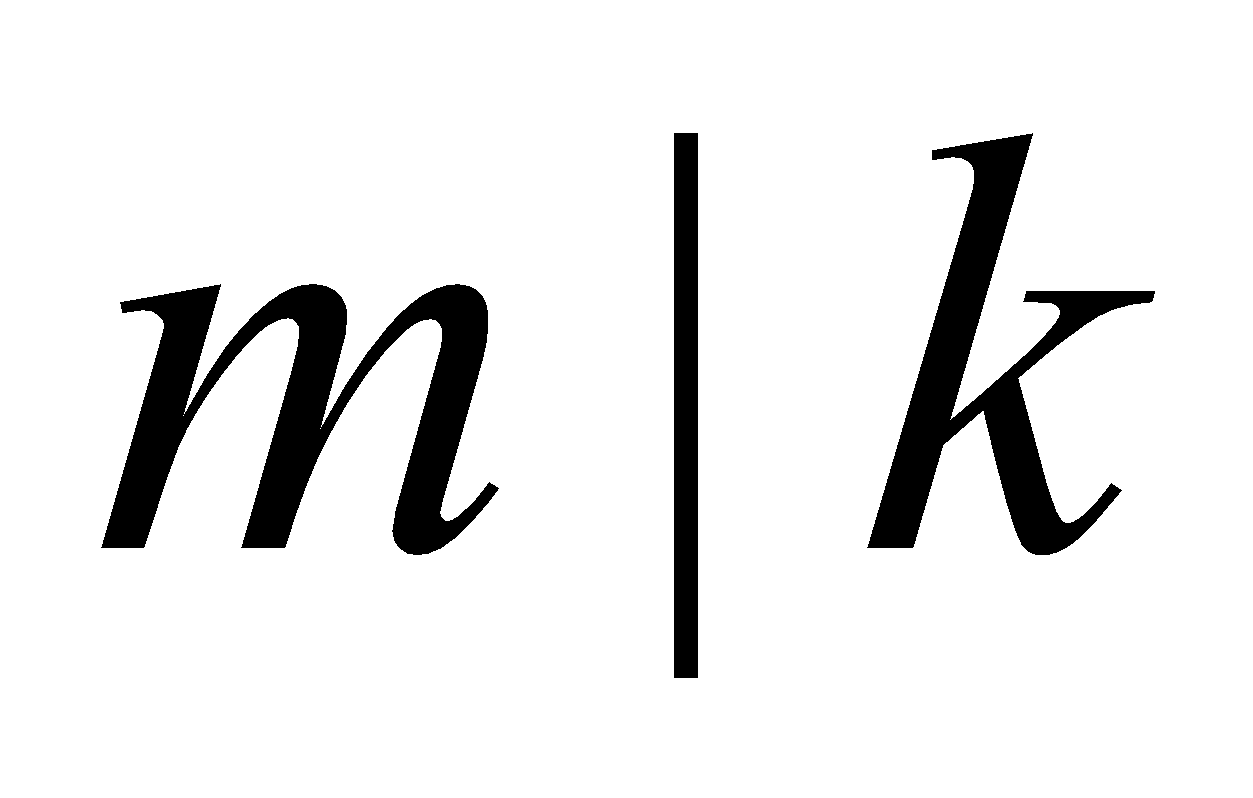
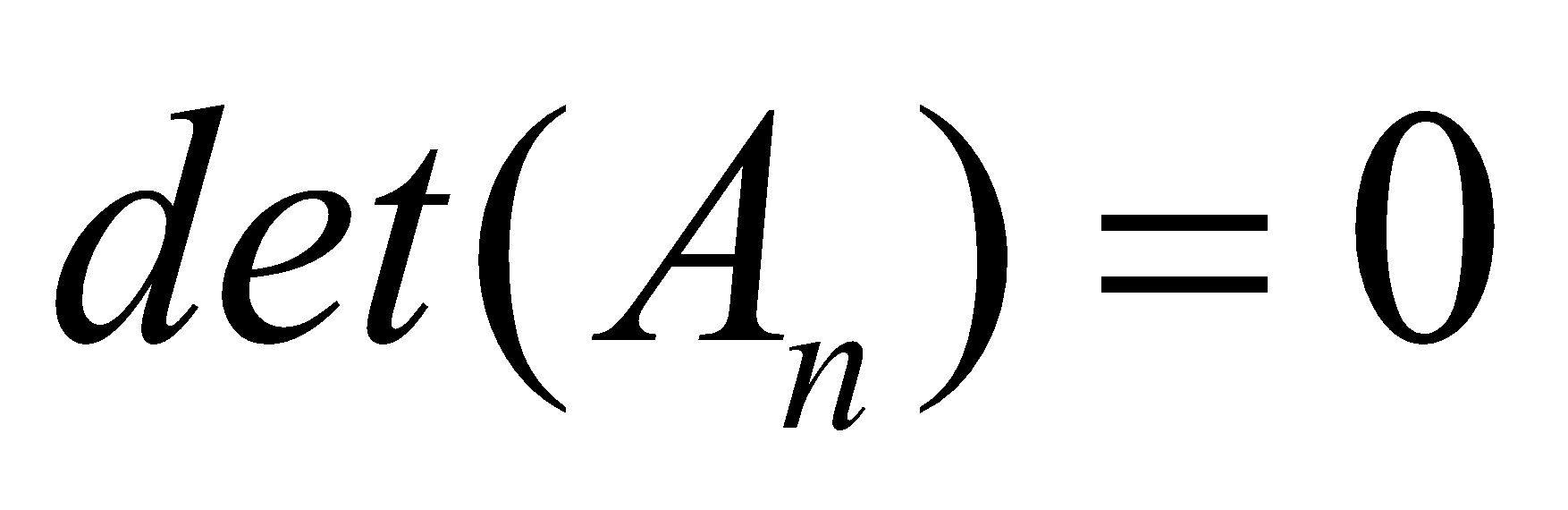
Случай 1.1  и 

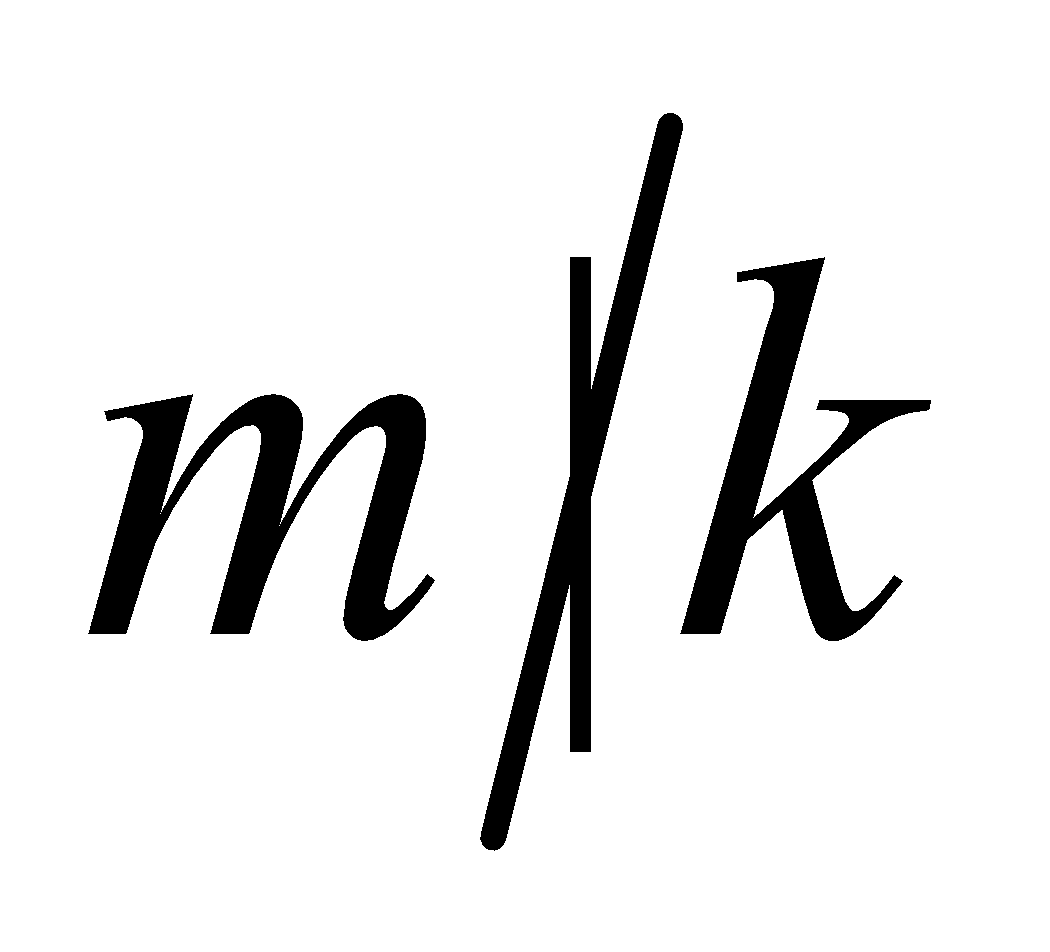
Мы рассмотрели случай 2.1, определенный выше.

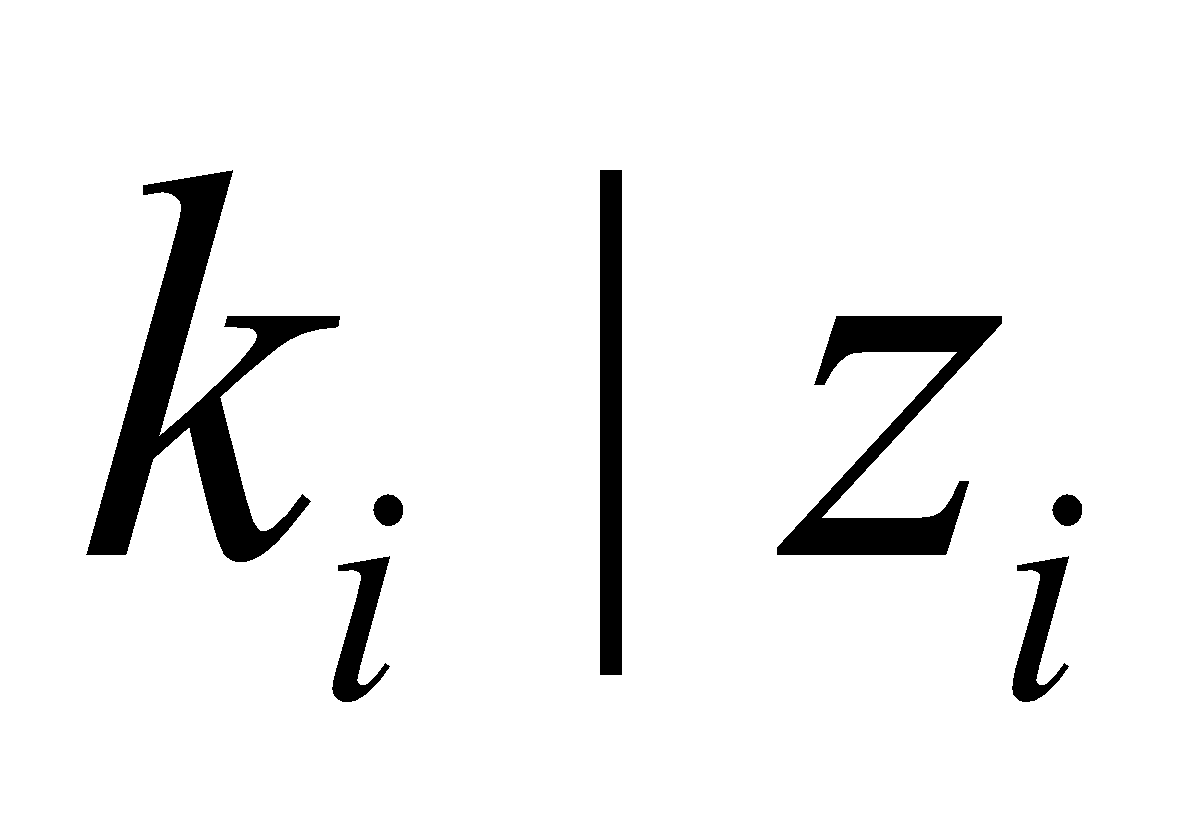
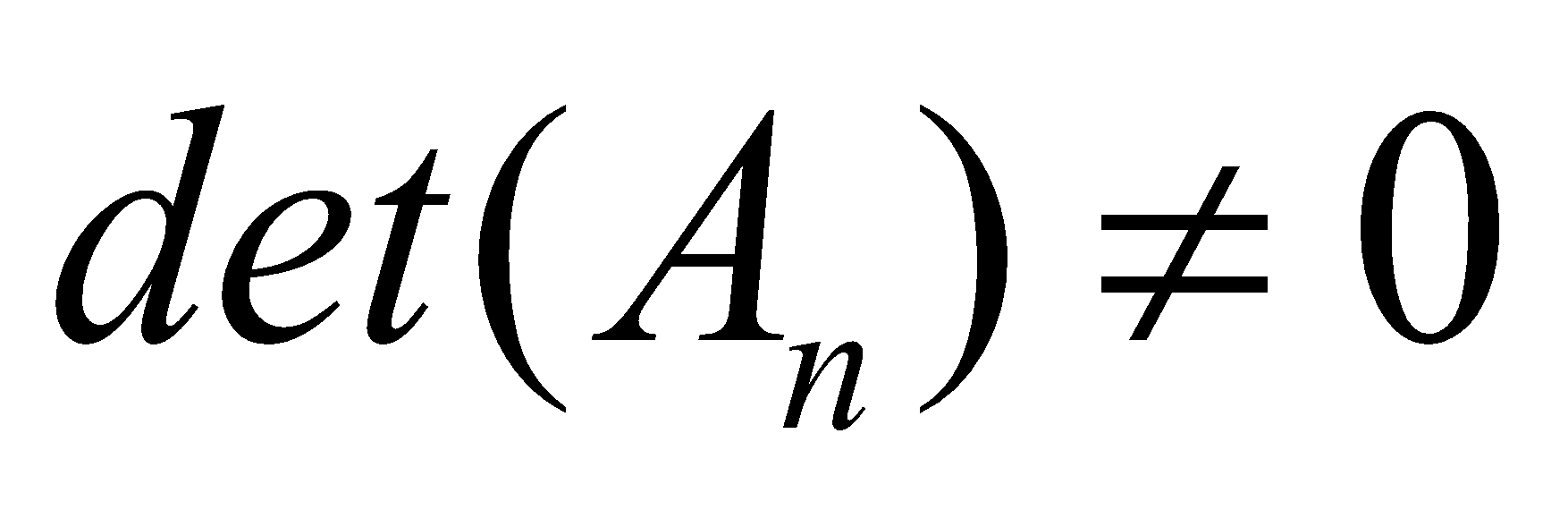
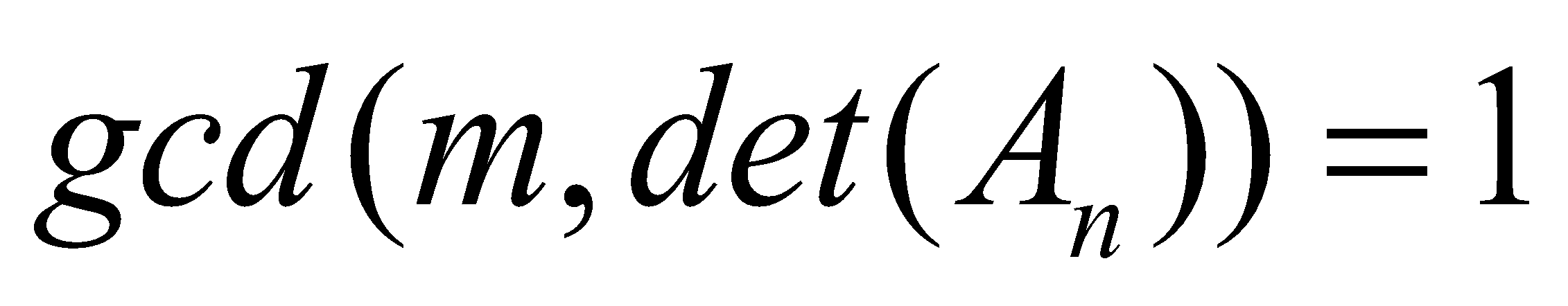
Случай 1.2  и 

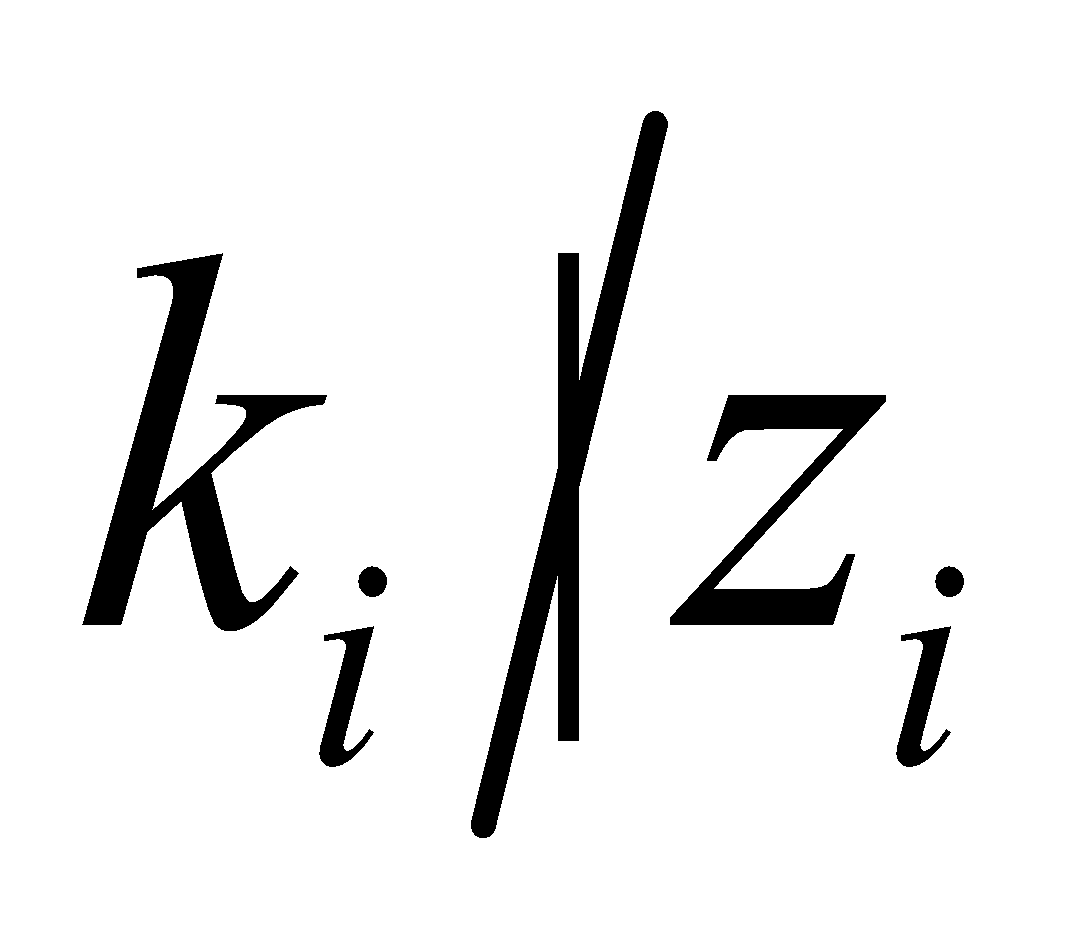
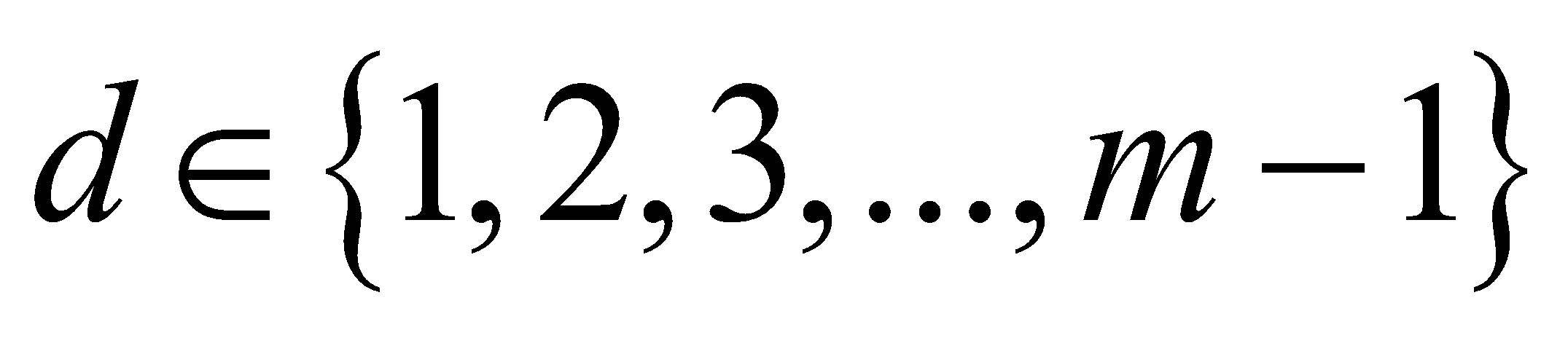
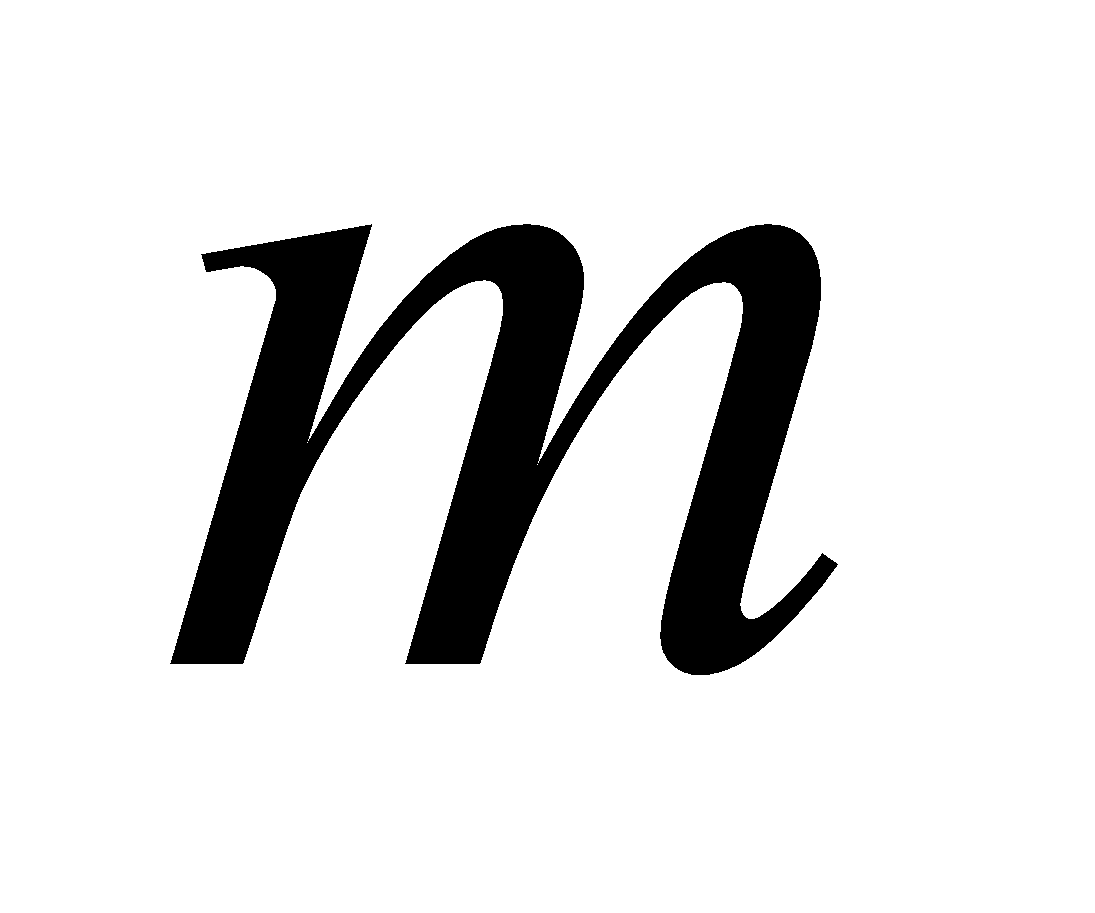
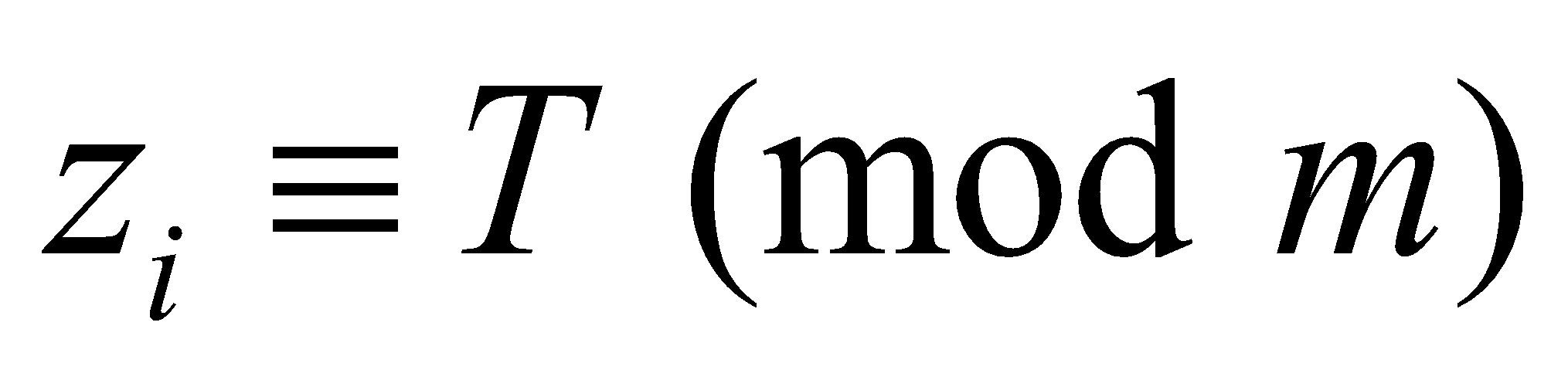
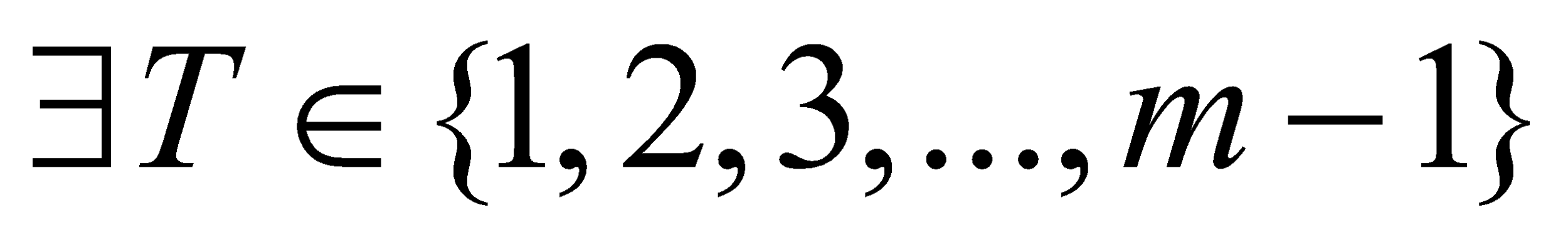
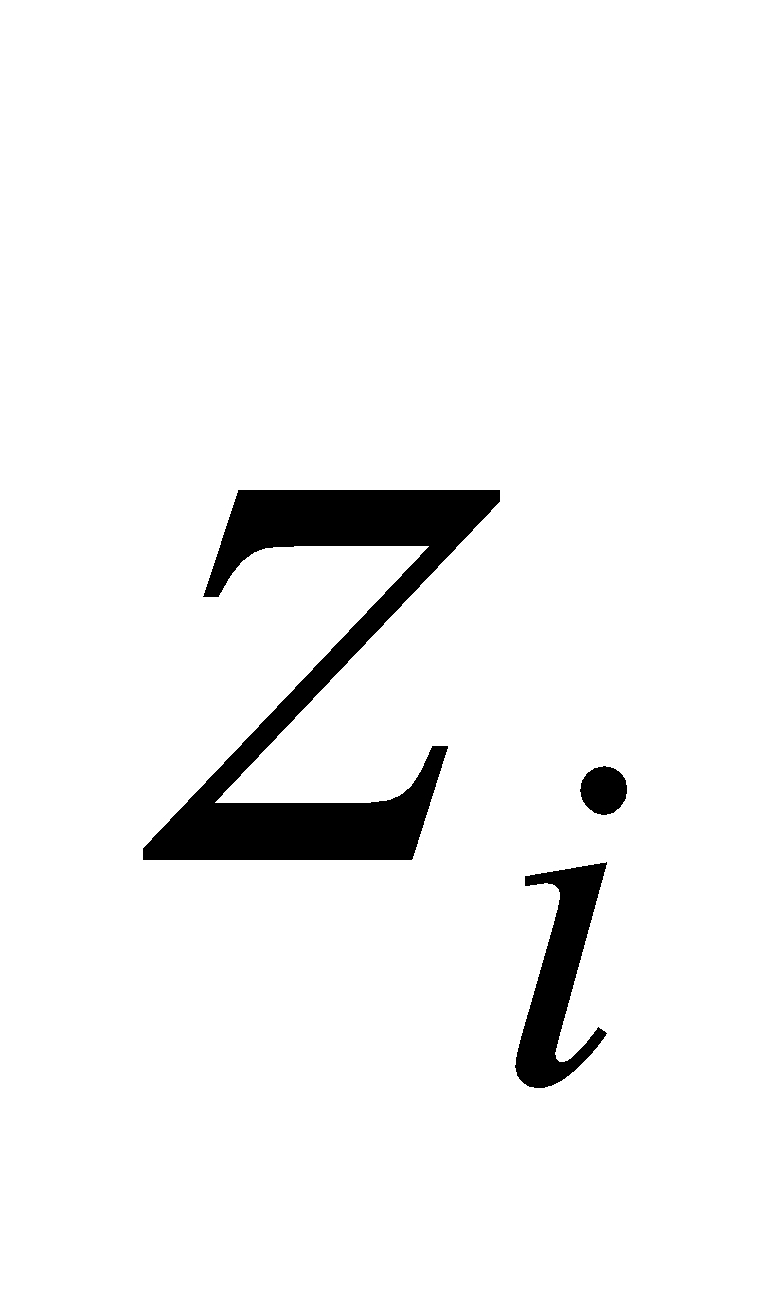
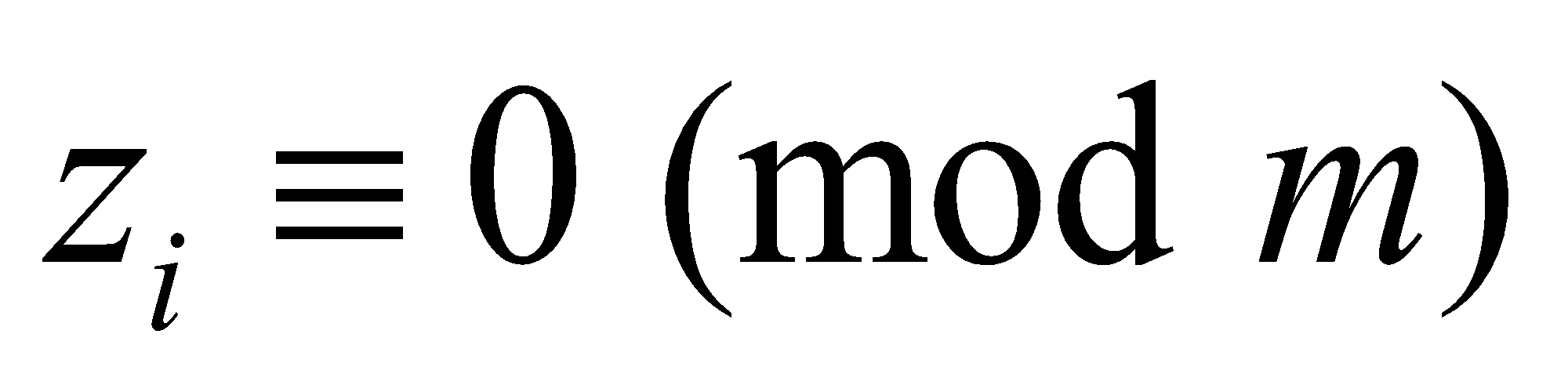
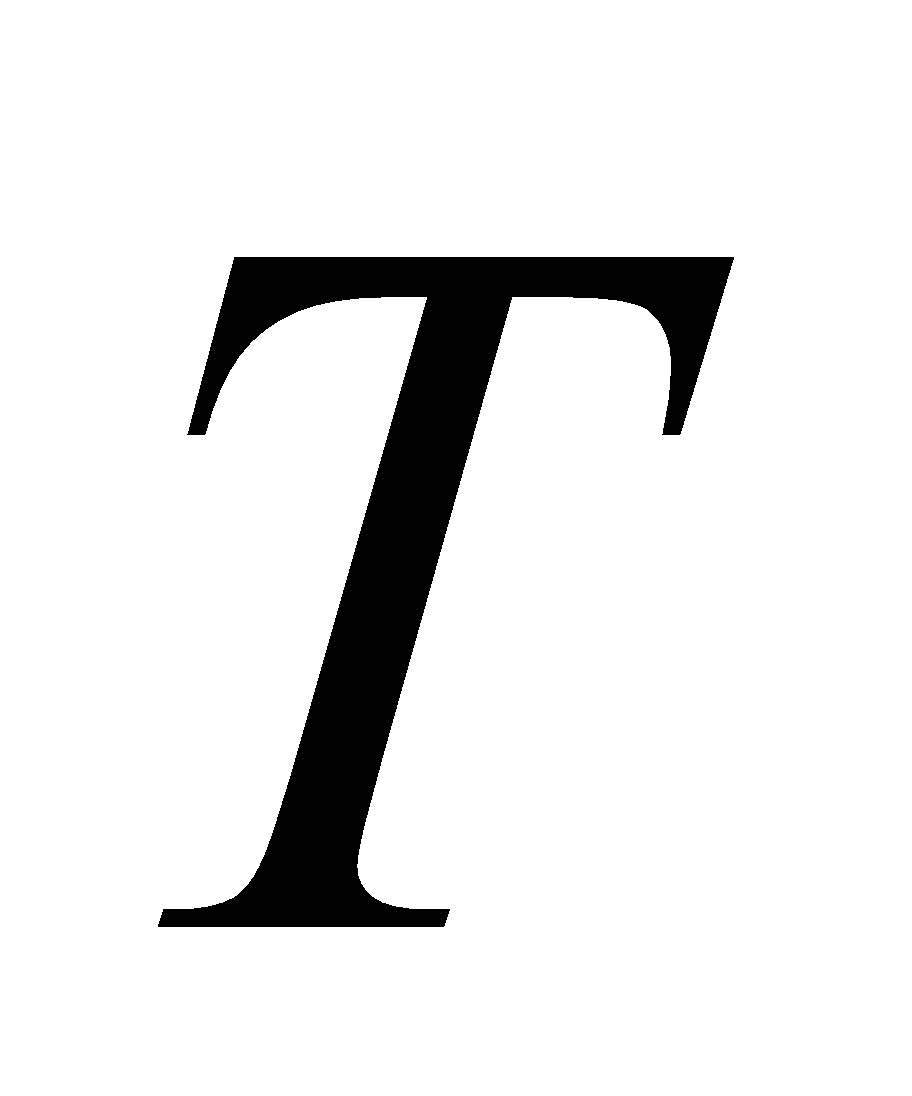
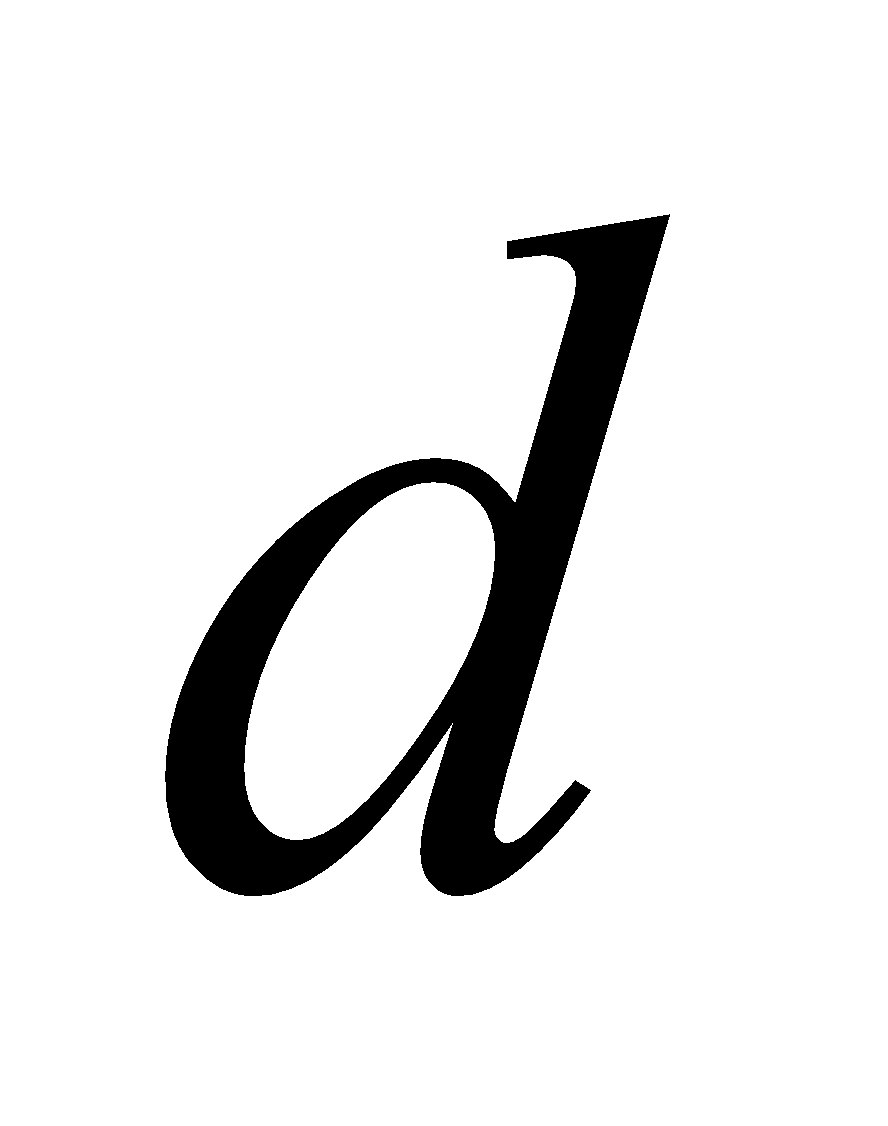
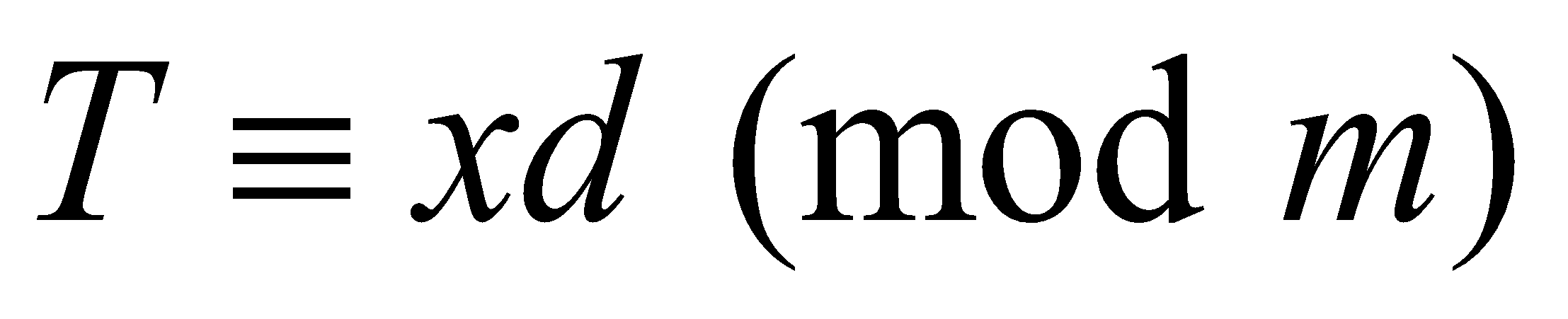
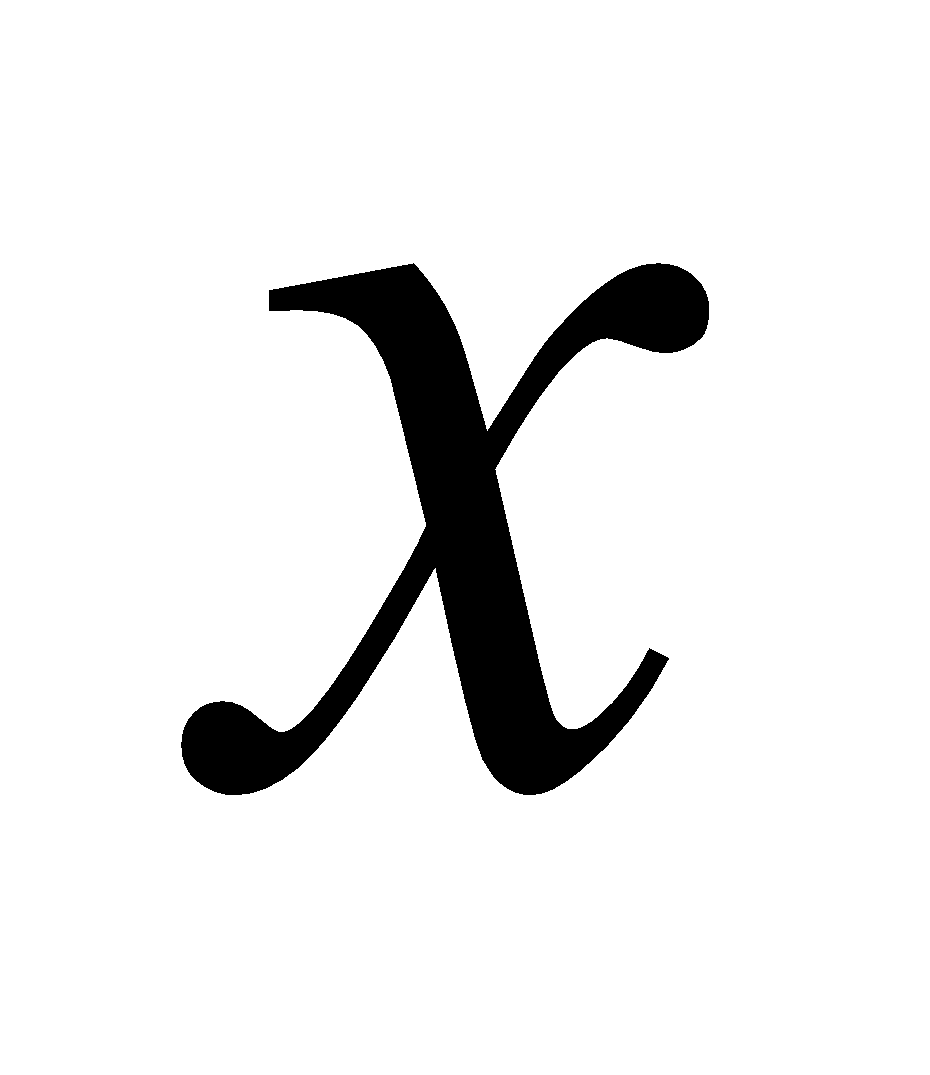
При сведении , мы можем предположить, что существует хотя бы одно i, такое что . Таким образом, мы не можем найти мультипликативно обратное к  по модулю .

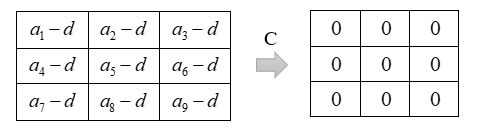
Таким образом, мы можем выделить два случая, а именно:

1.  Этот случай похож на Случай , мы обсудим его позже.

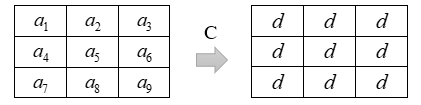
2.  Мы можем разделить этот случай на два частных случая, а именно:

2.1  Для этого случая мы можем выполнить действия, схожие со Случаем 2.1  и , тогда мы сможем легко найти ответ.

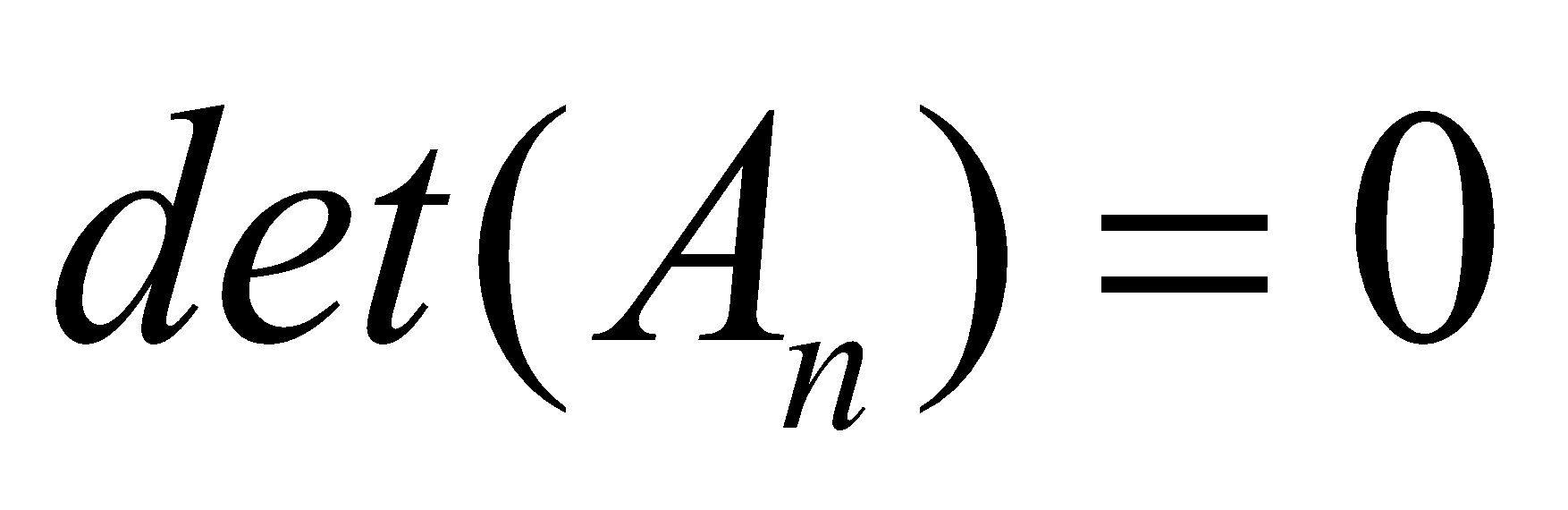
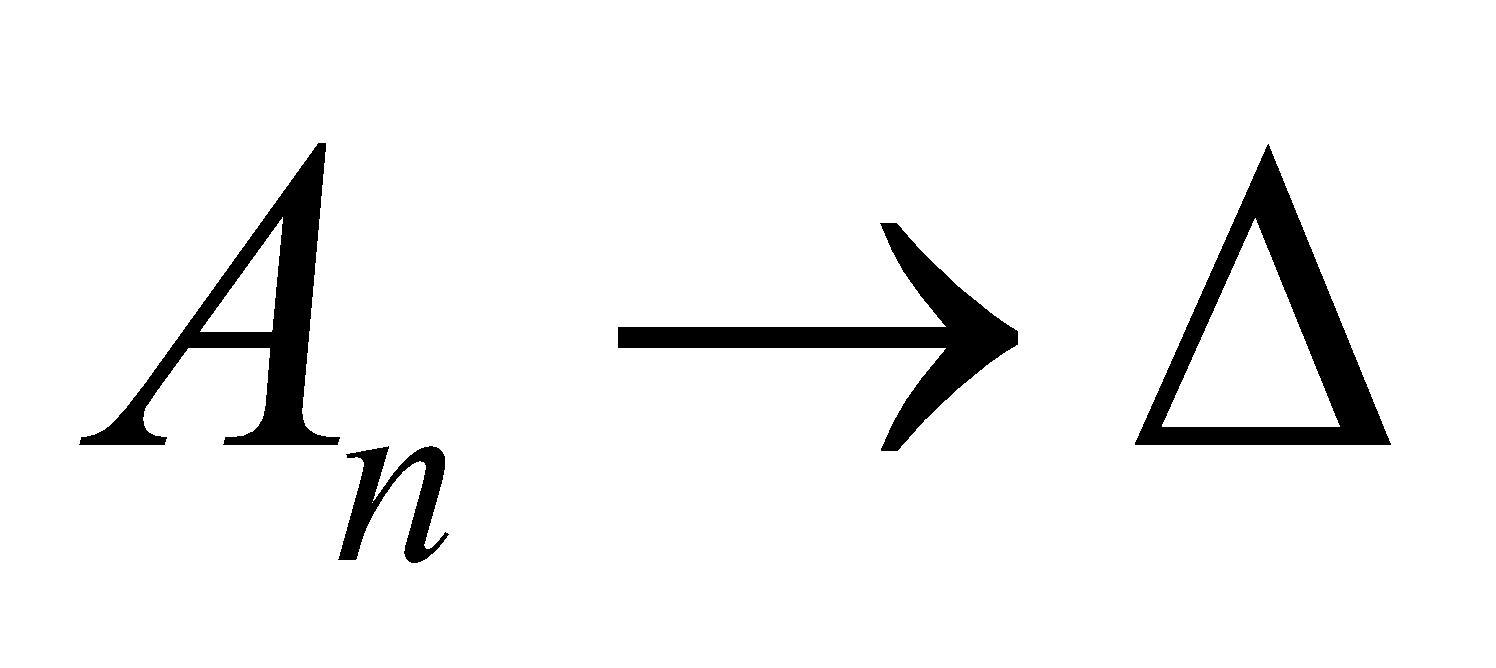
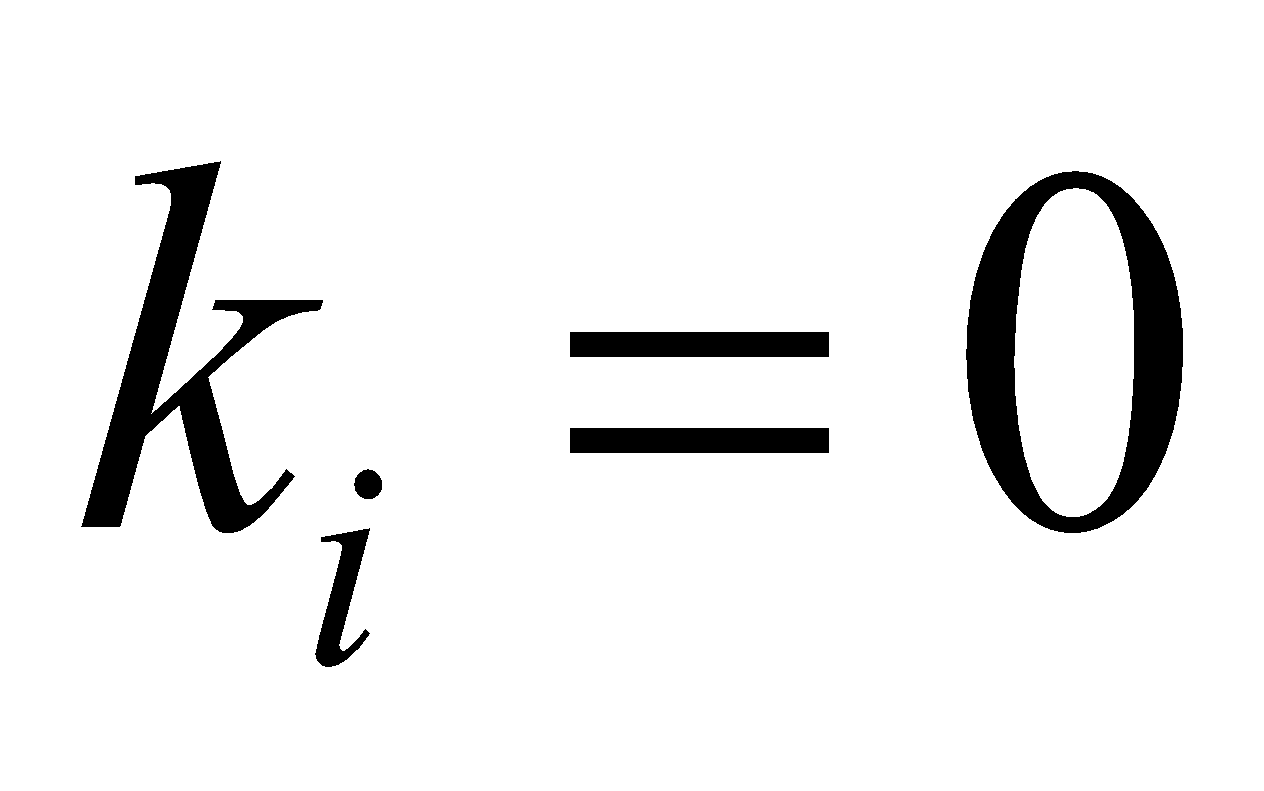
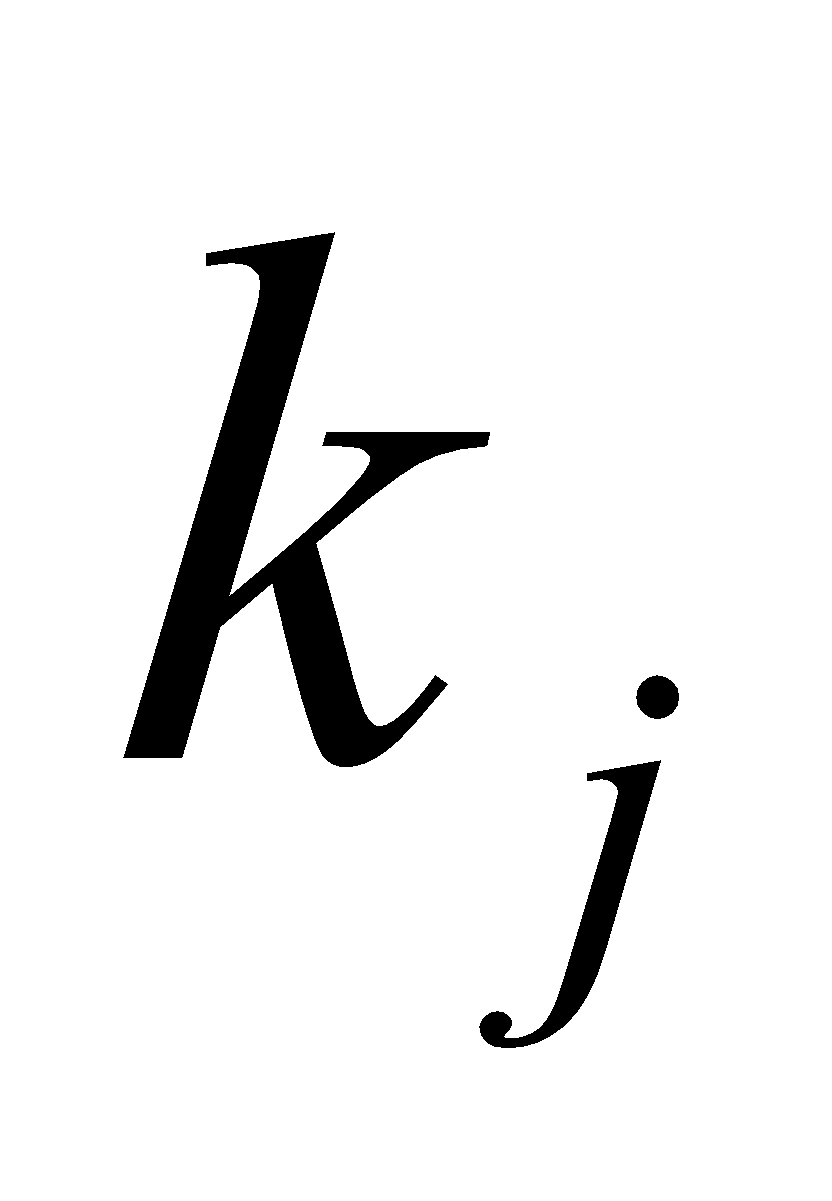
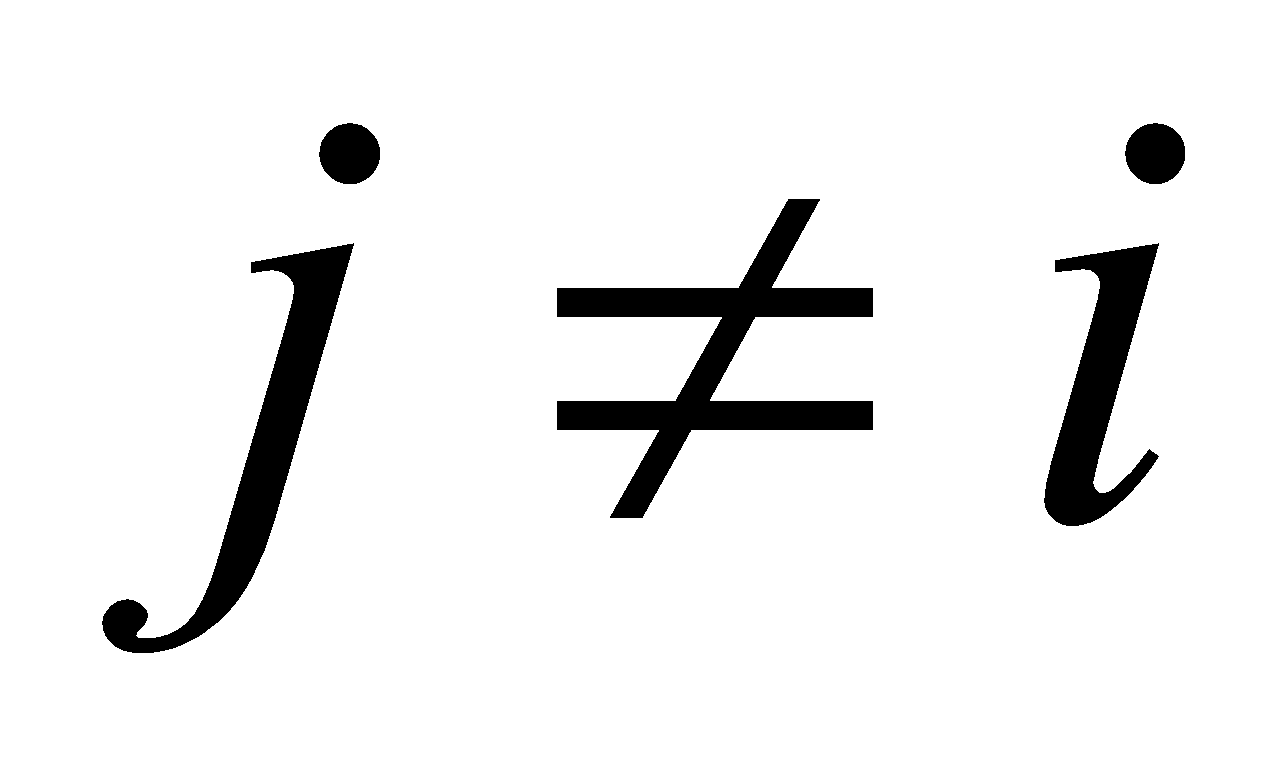
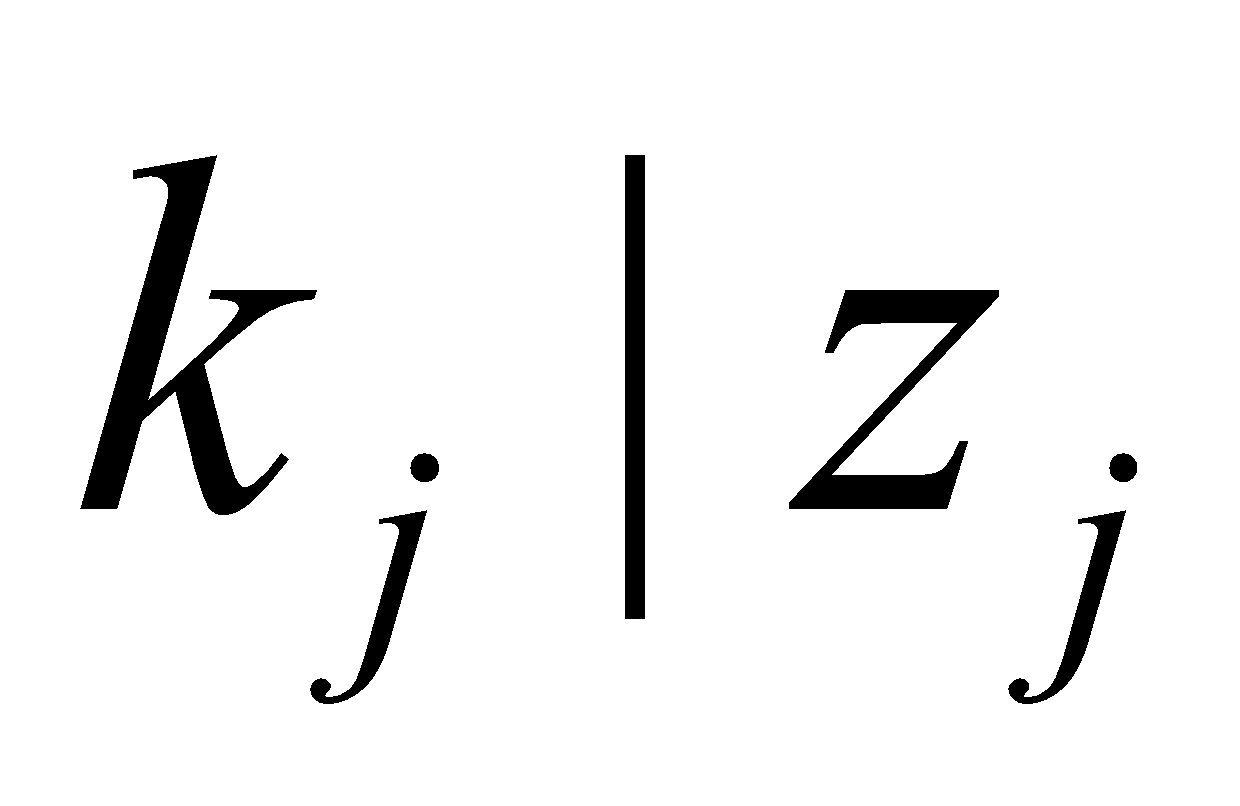
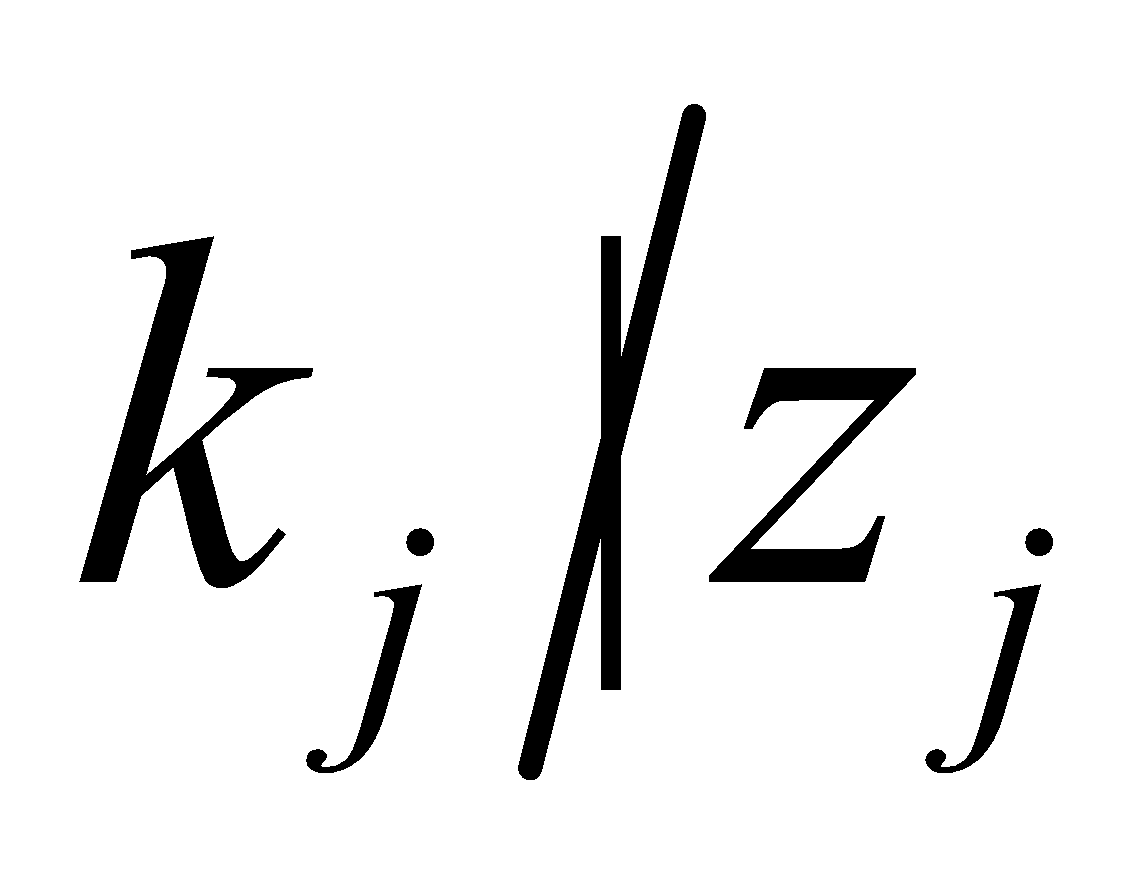
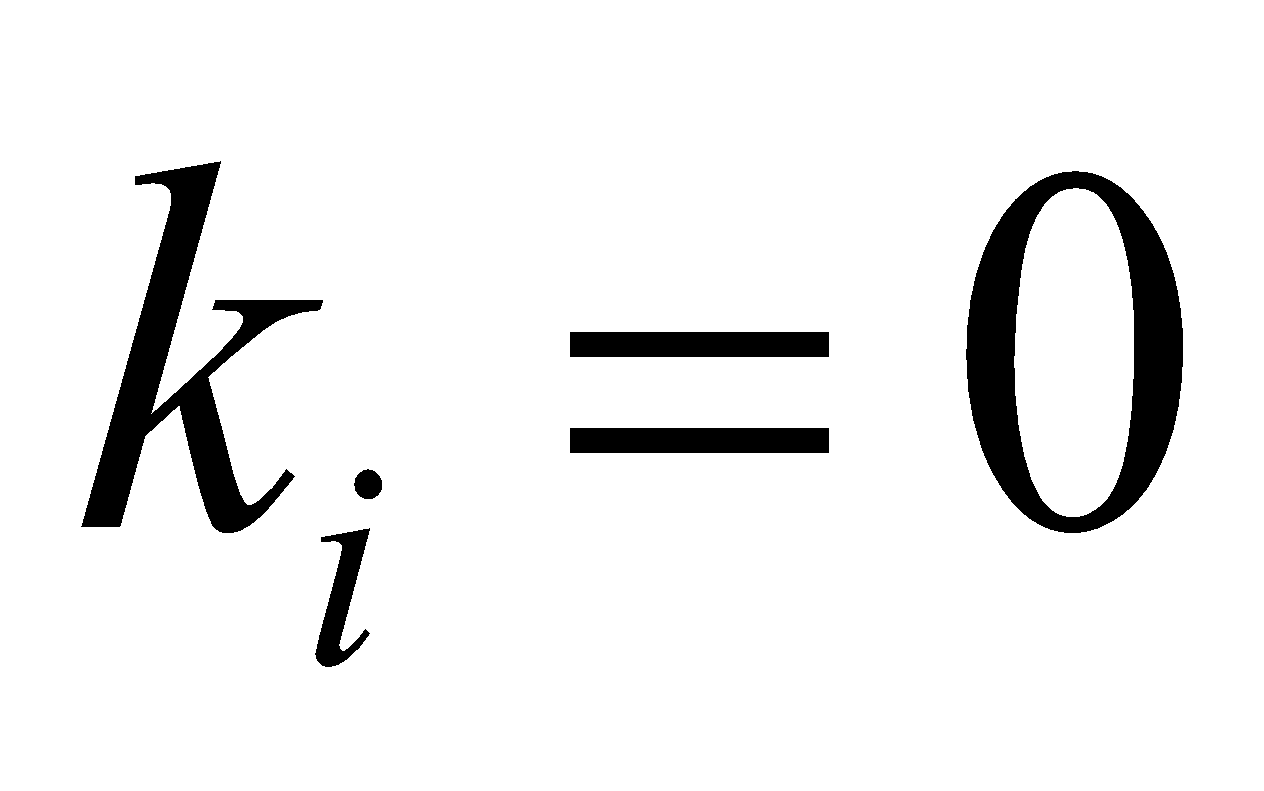
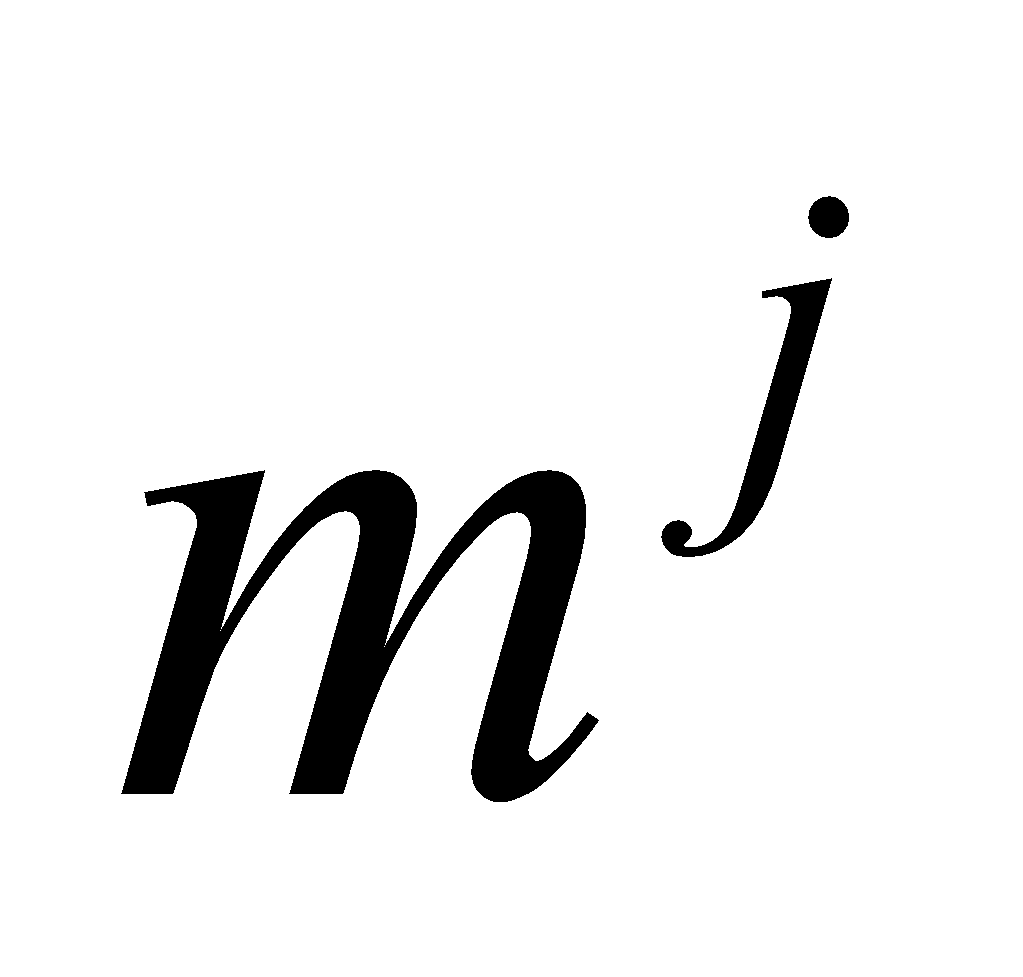
2.2  Этот случай показывает, что мы не можем найти ответ для начальной таблицы. Тогда мы поменяем таблицу на таблицу с d состояниями, где  и  это состояния начальной таблицы. Сперва мы предположим, что  . Мы исходим из того, что начальную таблицу можно заменить на таблицу с d состояниями. Таким образом мы получим новое  ,такое что  и найдем соотношение между  и  ,а именно  для некоторого целого . Мы можем отобразить это на рисунке.

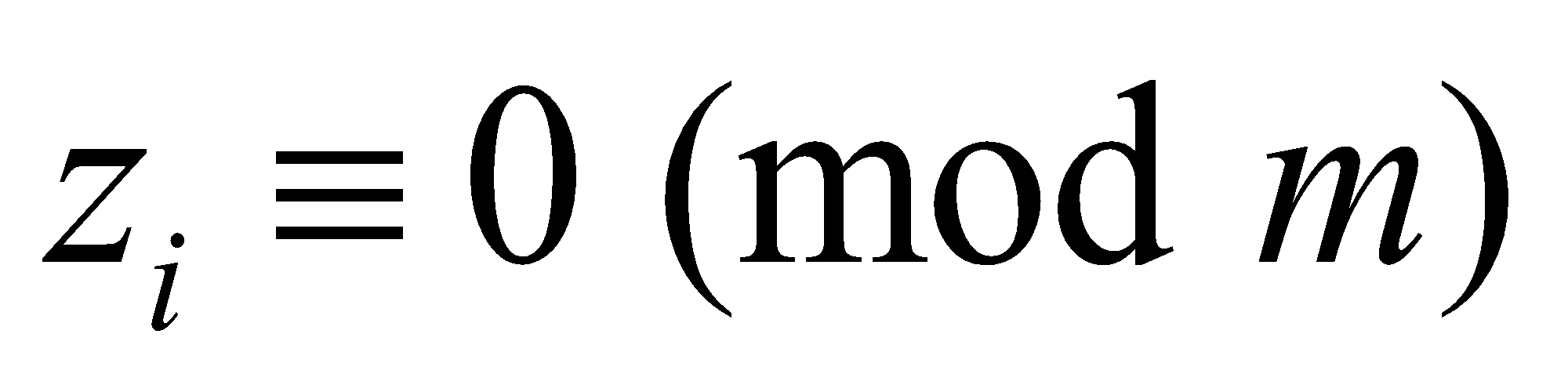
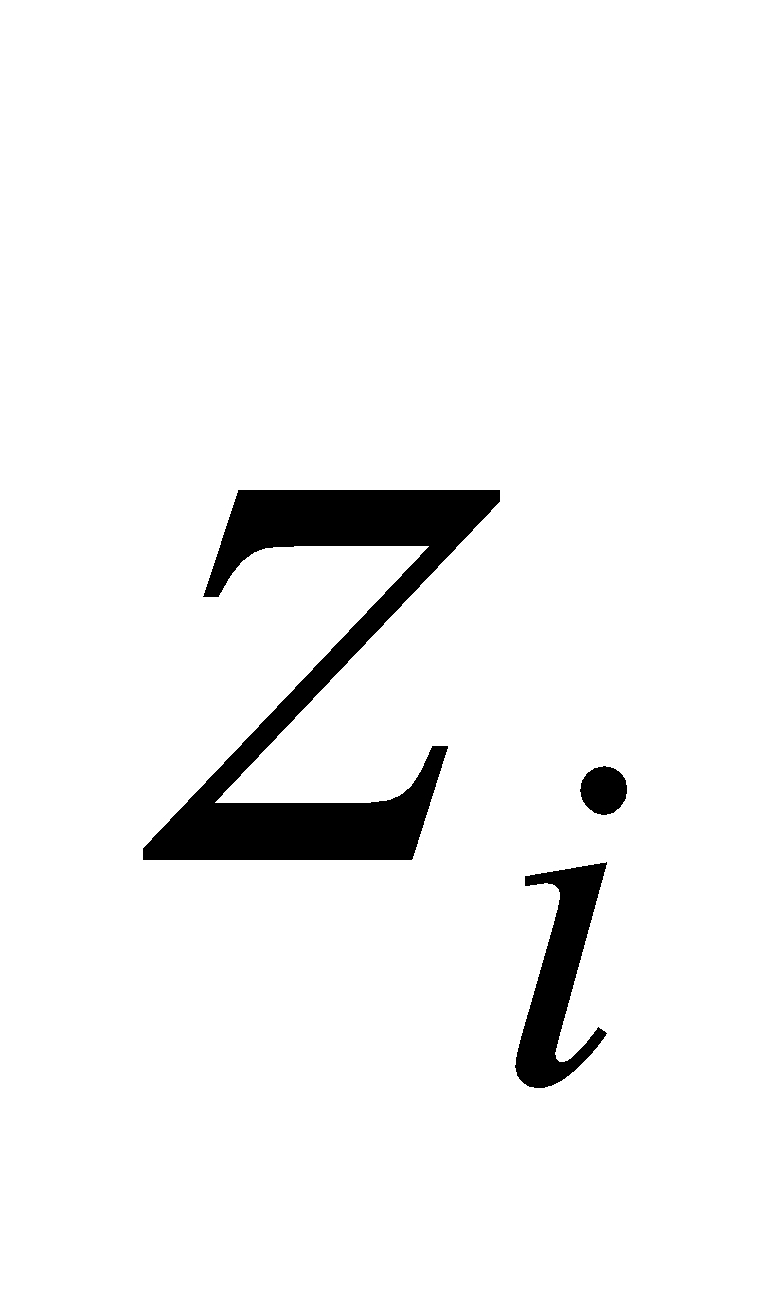
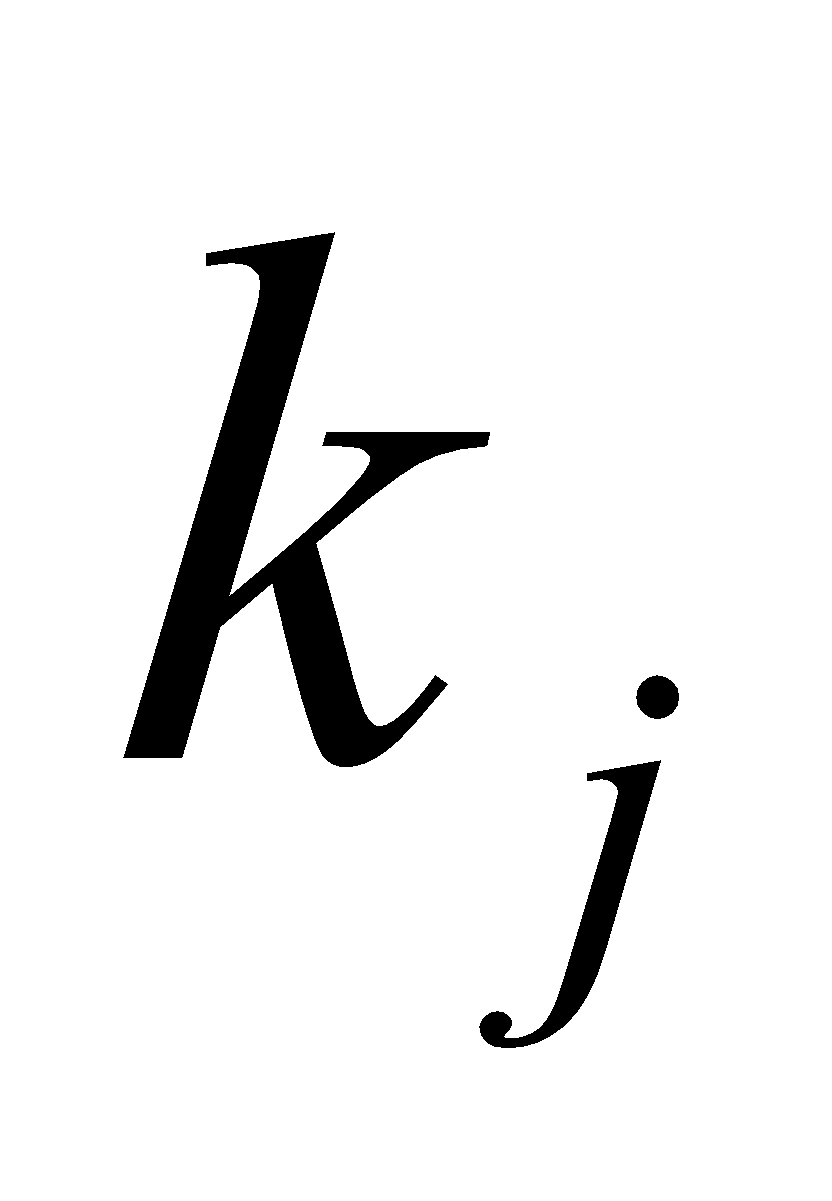
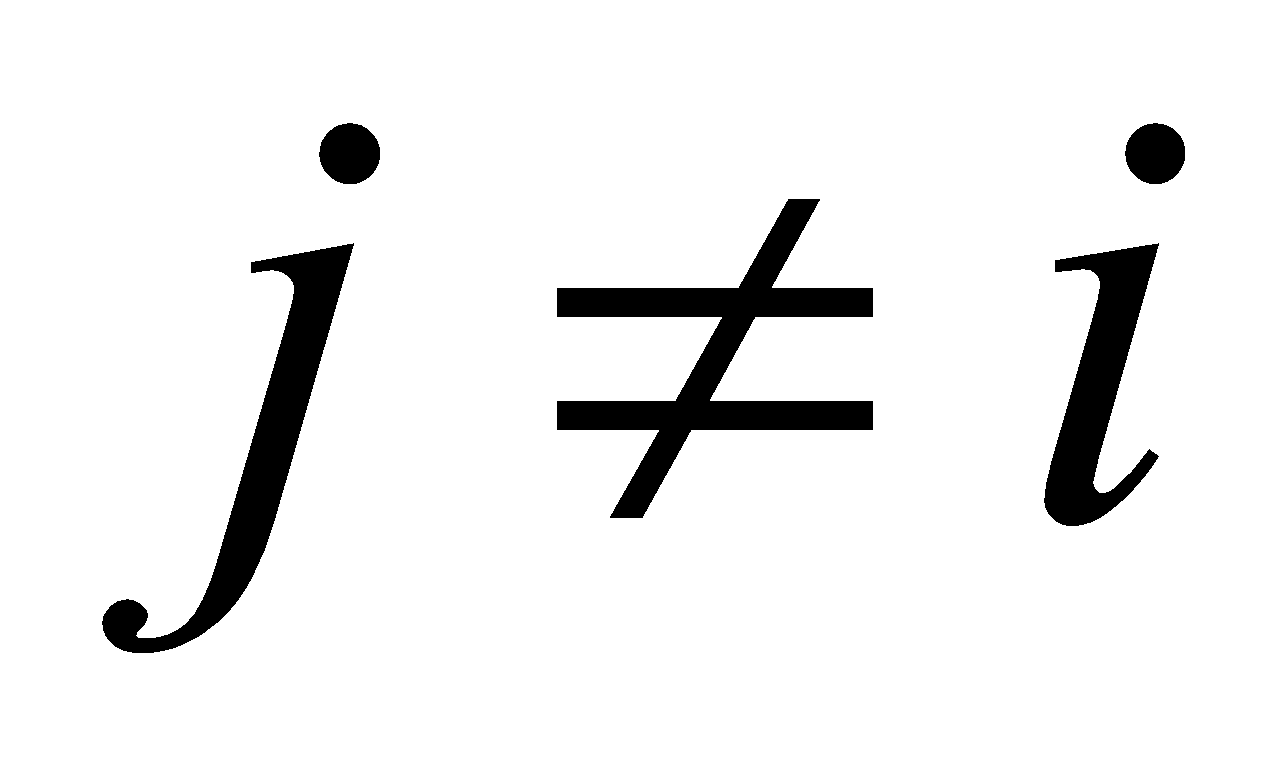


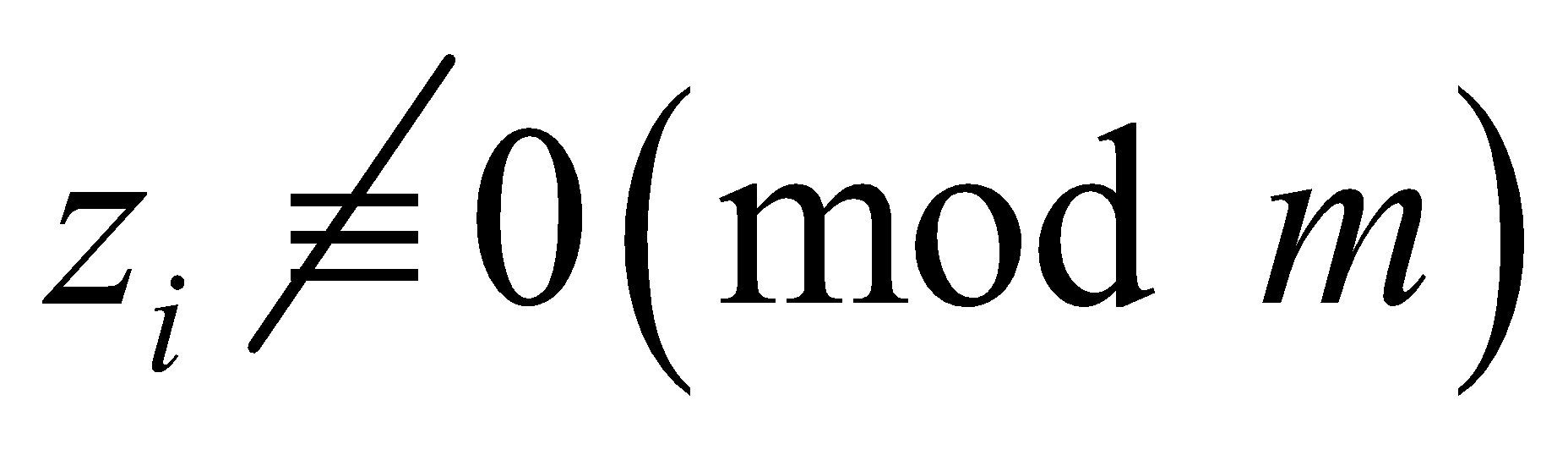
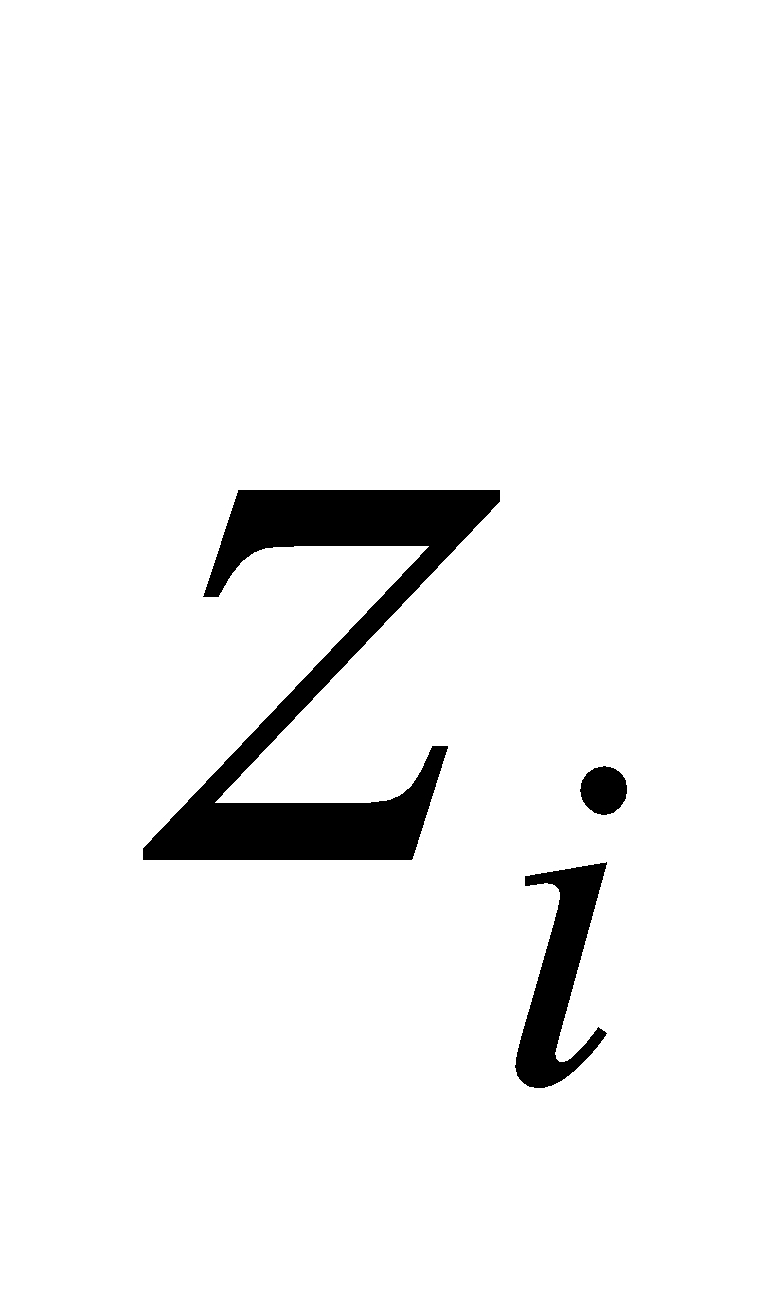
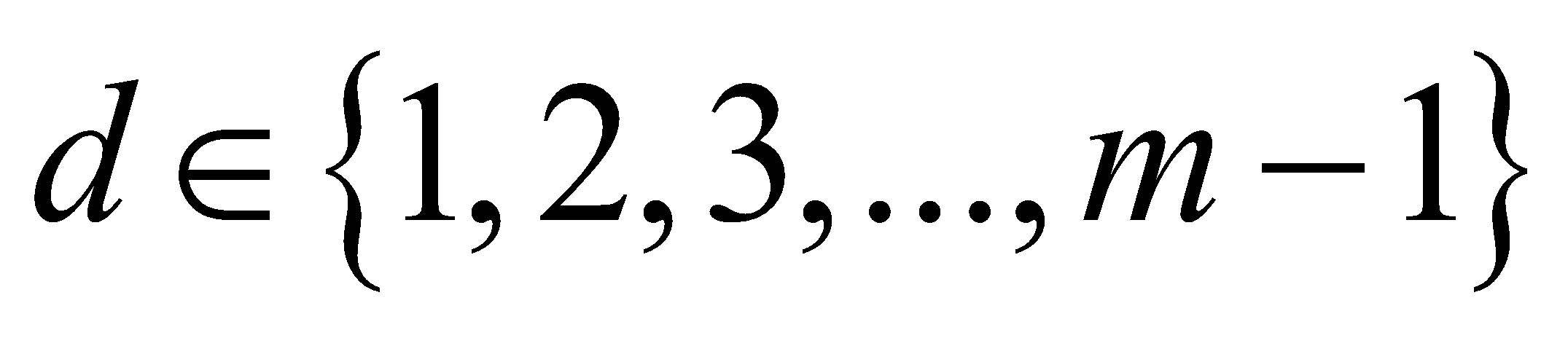
Таким образом

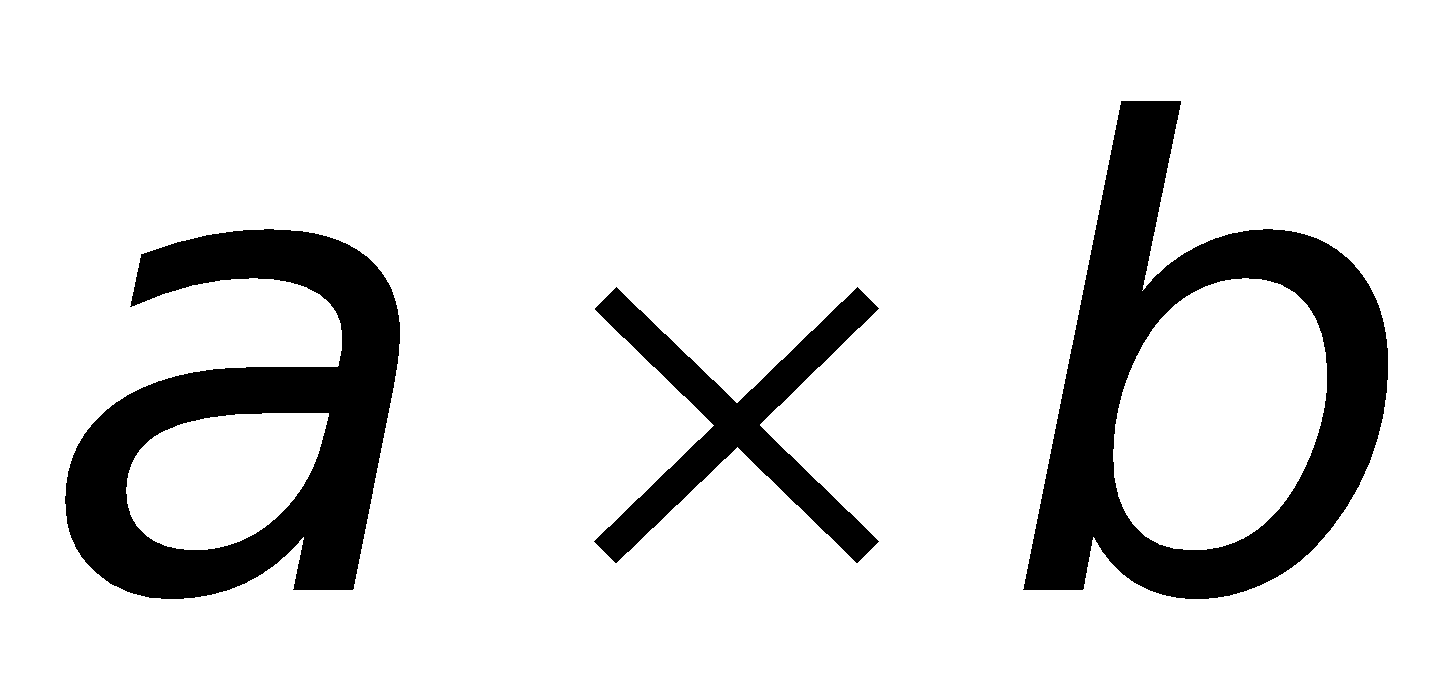
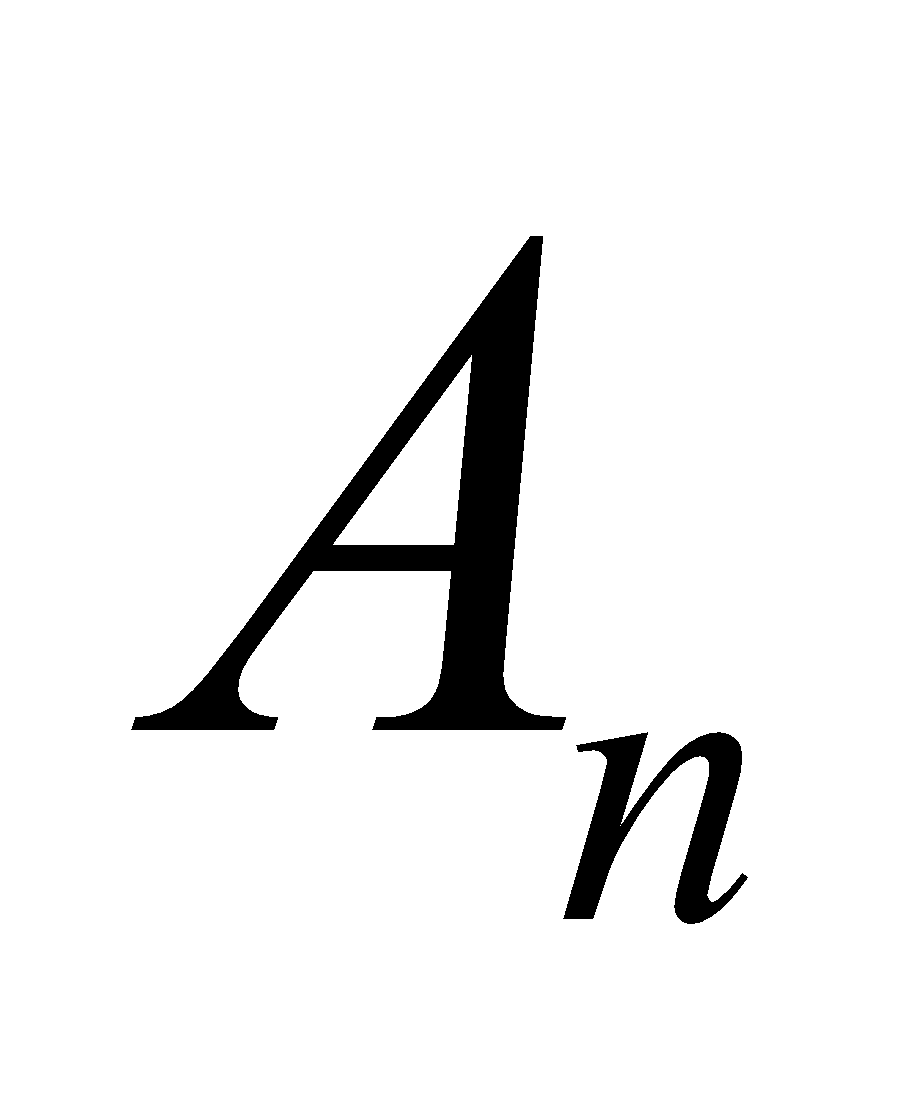
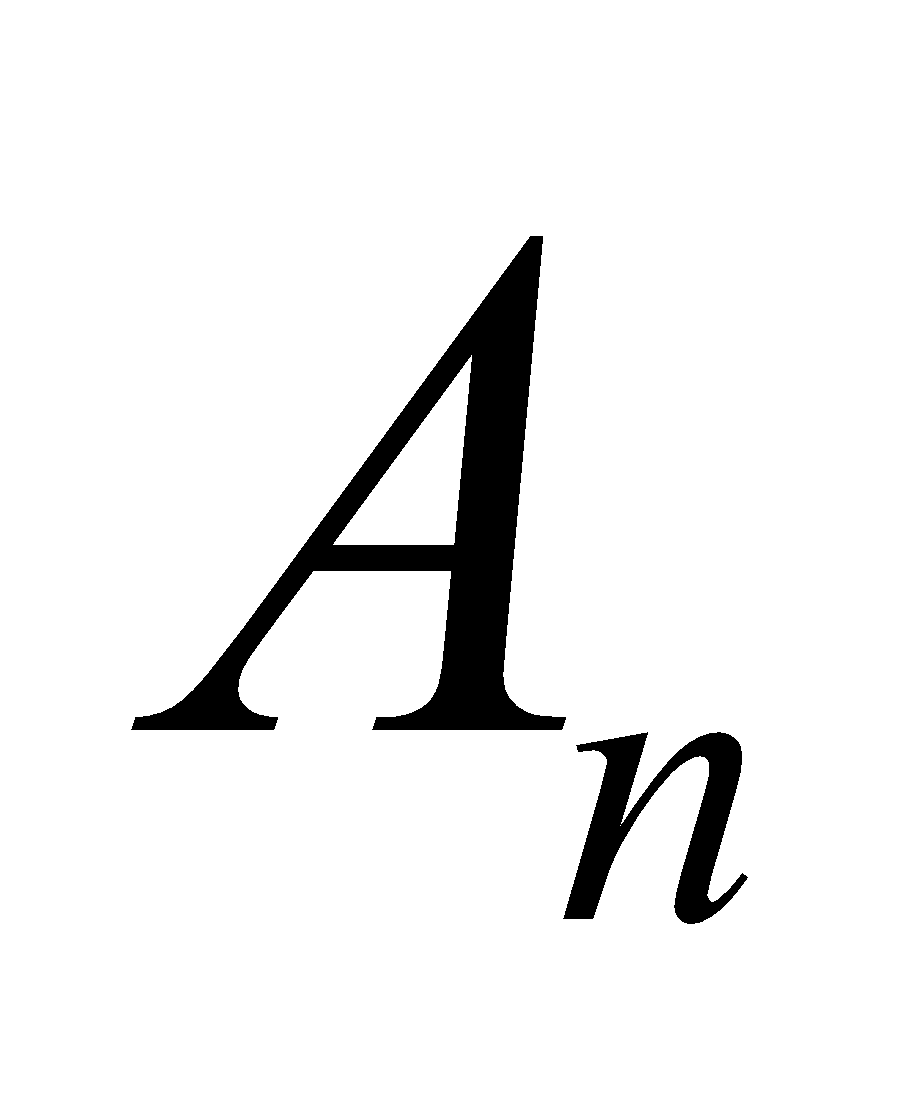
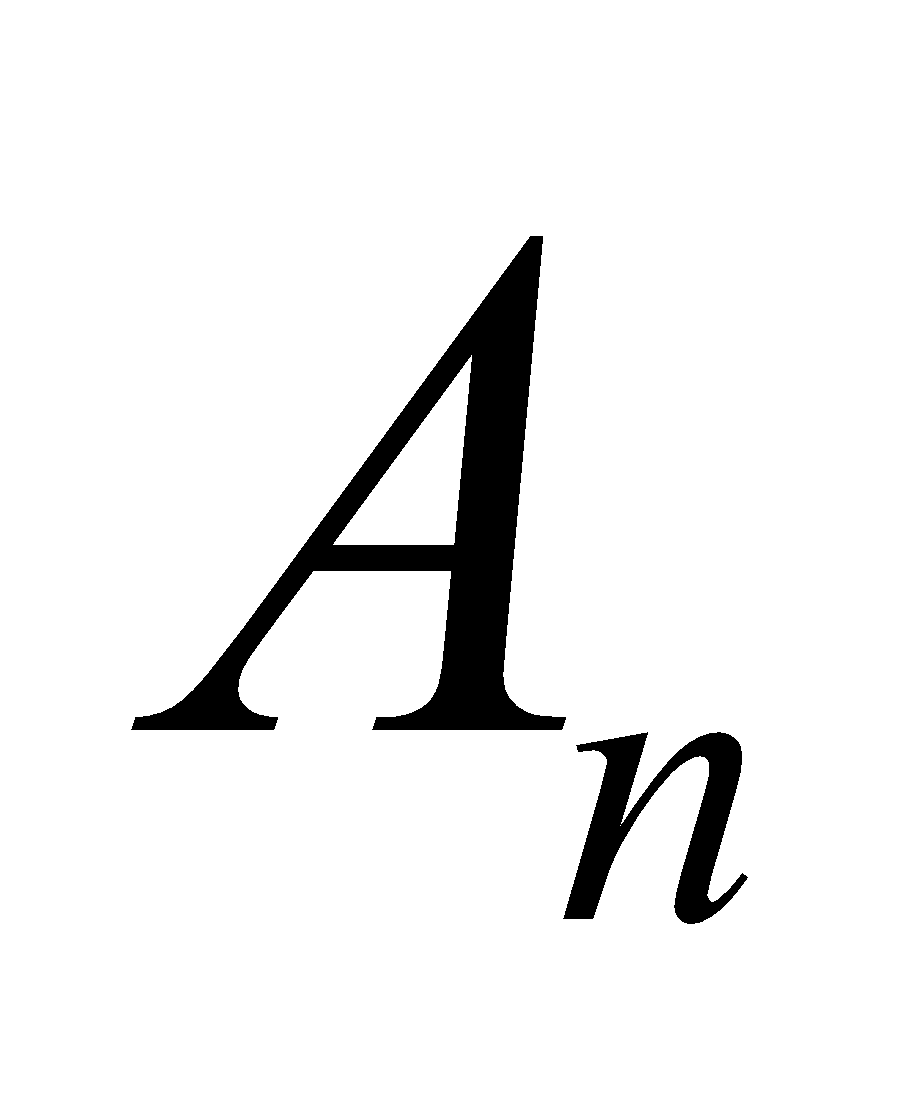


и мы найдем ответ (C), если он существует.

Случай 2  При сведении , мы можем предположить, что существует по меньшей мере одно i, такое что . Для других , , есть 2 возможных случая: , либо . Поэтому мы применяем тот же прием, что и выше. Поскольку при некотором i верно , то количество уравнений будет меньше, чем количество переменных. Таким образом имеется одна свободная переменная. Следовательно, у нас будет много ответов, а именно количество ответов будет равно  , где m это состояние начальной таблицы и j это количество свободных переменных. После этого мы выделим два случая, а именно:

1.  Следовательно, для  уравнение выполнено, и мы можем найти ответ рассматривая другие ,  так же, как и раньше.

2.  Следовательно, для  уравнение не выполнено. В этом случае мы можем заменить начальную таблицу на таблицу с d состояниями, где  как было указано выше.

Все эти методы поиска ответа годятся для каждой квадратной таблицы и любого состояния, в то время как целью данного проекта является нахождение ответа для любой таблицы с любыми состояниями и любой поверхностью, при условии что меняется состояние и нажатой лампочки, и соответствующих ей соседних лампочек. Действительно, мы можем использовать тот же самый метод для нахождения ответа. Мы должны лишь заменить  , поскольку оно ассоциировано с правилом нажатия и размером таблицы. Таким образом, другое правило нажатия на лампочку повлияет на матрицу , а размер таблицы повлияет на размер .

Разработчики также создали программу для проверки методов, и эта программа также может найти решения. Эти знания можно использовать в повседневной жизни, например: 1) при поливе растений, так как в наши дни люди устанавливают спринклеры не учитывая то, какой объем воды требуется для каждого растения. Мы можем представить сад в виде таблицы и записать единицы воды, которые нужны растениям в каждой ячейке таблицы. После этого мы используем данные, полученные в этом проекте для установки спринклеров. Таким образом, растения получат тот объем воды, в котором они на самом деле нуждаются. Это поможет сэкономить деньги и воду. 2) Улучшение сенсорных датчиков, так как в настоящее время есть некоторые датчики, которые имеют высокую вероятность ошибки, что может быть вредным или неэффективным. Следовательно, мы поможем повысить их эффективность, представив помещение в виде таблицы, установив области, которые закреплены за каждым датчиком. Затем мы будем использовать результаты этого проекта, чтобы установить датчики так, чтобы каждую область проверяли 3 датчика, и позволить датчикам голосовать, есть ли в этой области люди или нет. Мы надеемся, что эти знания помогут изобрести что-то новое и принесут пользу миру.

[1]Eisele, R. (2018). LightsOut Solution using Linear Algebra. Retrieved from https://www.xarg.org/2018/07/lightsout-solution-using-linear-algebra/

[2]Weisstein, E. W. (n.d.-a). Lights Out Puzzle. *MathWorld*. Retrieved from <http://mathworld.wolfram.com/LightsOutPuzzle.html>