# Задача № 1: Телевизионная игра «Колесо Фортуны»

**Спецификация задачи:**

**Список возможных действий m=:**

**Список возможных событий n=:**

Примечание: Для этой задачи существуют всего nхm=nm возможных комбинаций Действие–Событие.

# Критерии оценки эффективности решения задачи

Таблица вида А-1

Примечание: Этот вариант таблицы А-1 требует **квалифицированного пользователя**, который имеет предварительные знания о том, что такое прибыль и что прибыль может быть отрицательной величиной.

Таблица вида А-1. Version N

Примечание: Здесь данные платежной таблицы из спецификации задачи трактуются как Премия и Штраф в зависимости от знака числа. Этот вариант таблицы А-1 не требует высокой квалификации пользователя, который должен иметь только лишь простые базовые знания о том, что такое премия и что такое штраф.

# Матрица решений (для одного показателя – прибыль)

Таблица. Матрица решений (Исходные данные задачи)

# Система обозначений для использования в описании алгоритмов решений задачи

**n, m —** это количество событий и количество действий в данной задаче. Например, в Задаче №1: n=2; m=2;

**Act\_j** это действие под номером j (j=1,2,…,m), которое для каждой конкретной задачи имеет свое наименование. Например, в Задаче №1: Act\_1 – Нажать на кнопку сильного вращения; Act\_2 – Нажать на кнопку слабого вращения. (Действия на любом языке наиболее адекватно выражаются ***повелительным//указательным*** предложением, начинающимся глаголом).

**Event\_i** это событие под номером i (i=1,2,…,n), которое для каждой конкретной задачи имеет свое наименование. Например, в Задаче №1: Event\_1 – Колесо остановится в белой области; Event\_2 – Колесо остановится в красной области. (События на любом языке наиблоее адекватно выражаются ***повествовательным//описательным*** предложением);

**P(Event\_i)** это значение вероятности появления события Event\_i, где i=1,n. Например, в Задаче №1: P(Event\_1)=0.3; P(Event\_1)=0.7;

**CP(Event\_i | Act\_j)** это числовое значение условной прибыли при данной комбинации события Event\_i и действия Act\_j, где i=1,n; j=1,m. Например, в Задаче №1: CP(Event\_1|Act\_1)=$800; CP(Event\_2|Act\_1)=$200; CP(Event\_1|Act\_2)= –$2500; CP(Event\_2|Act\_2)=$1000;

**WP(Event\_i | Act\_j)** это числовое значение взвешенной прибыли при данной комбинации события Event\_i и действия Act\_j, где i=1,n; j=1,m;

**EMV (Act\_j)** это среднее ожидаемое денежное значение при выборе действия Act\_j, где j=1,m;

**COL(Event\_i | Act\_j)** это значение условной потери при данной комбинации события Event\_i и действия Act\_j, где i=1,n; j=1,m;

**CPmax(Event\_i)** это максимальное значение условной прибыли для данного события Event\_i, где i=1,m;

**WOL(Event\_i | Act\_j)** это значение взвешенной потери при данной комбинации события Event\_i и действия Act\_j, где i=1,n; j=1,m;

**EOL(Act\_j)** это значение средней ожидаемой потери при выборе действия Act\_j, где j=1,m;

# Вычислительный алгоритм для решения задачи по критерию «Средняя ожидаемая прибыль»

Процесс решения проблемы по критерию «**Средняя ожидаемая прибыль**» состоит из вычислений по следующим 4 алгоритмам:

1. Алгоритм №1. Вычисления значений взвешенной прибыли
2. Алгоритм №2. Вычисление среднего ожидаемого денежного значения
3. Алгоритм №3. Определение максимальной величины среднего денежного значения
4. Алгоритм №4. Определение Действия (или Действий), которое является решением задачи по критерию среднего ожидаемого денежного значения

## Описание процесса решения проблемы:

Расчеты по алгоритму №1:i=1; j=1: WP(Event\_1|Act\_1)= CP(Event\_1|Act\_1)\* P(Event\_1)=   
i=1; j=2: WP(Event\_1|Act\_2)= CP(Event\_1|Act\_2)\* P(Event\_1)=   
i=2; j=1: WP(Event\_2|Act\_1)= CP(Event\_2|Act\_1)\* P(Event\_2)=   
i=2; j=2: WP(Event\_2|Act\_2)= CP(Event\_2|Act\_2)\* P(Event\_2)=

Расчеты по алгоритму №2:j=1: EMV(Act\_1)= WP(Event\_1|Act\_1)+ WP(Event\_2|Act\_1)=

j=2: EMV(Act\_2)= WP(Event\_1|Act\_2)+ WP(Event\_2|Act\_2)=

Расчеты по алгоритму №3:Max\_EMV=max{ EMV(Act\_1); EMV(Act\_2)}=max{}=

Расчеты по алгоритму №4:j=1: Так как EMV(Act\_1)= и Max\_EMV=, то Solution = Act\_1 “Нажатие на кнопку Сильное вращение”;  
j=2: Так как EMV(Act\_2)= и Max\_EMV=, то Solution ≠ Act\_2.

## Результаты вычисления по алгоритмам 1-4

Таблица. Средняя ожидаемая денежная прибыль.

## Рекомендация для клиента

Моему клиенту рекомендуется выбрать действие Act\_1 **“Нажать на кнопку Сильное вращение”**. Такое решение обеспечивает ему победу в этой игре с максимальной **средней ожидаемой прибылью** равной 380 долларов. Он обеспечит себе такое максимальное среднее ожидаемое значение прибыли, если он ***бесконечно долго*** будет играть в эту игру, и при этом ***каждый раз*** будет нажимать на ***кнопку сильного вращения*** при заданных в Таблице №3: Исходные данные задачи. Это значит, что минимальное среднее ожидаемое значение прибыли составит 380 долларов на одну игру.

# Вычислительный алгоритм для решения задачи по критерию «Средние ожидаемые потери»

1. Алгоритм №5. Вычисление максимального значения условной прибыли
2. Алгоритм №6. Вычисление значения условных возможных потерь
3. Алгоритм №7. Вычисление значения взвешенных возможных потерь
4. Алгоритм №8. Определение минимальной величины средних ожидаемых потерь.
5. Алгоритм №9. Определение минимальной величины средних ожидаемых потерь
6. Алгоритм №10. Определение Действия (или Действий), которое является решением задачи ТПР по критерию средних ожидаемых потерь

## Описание процесса решения проблемы

Расчеты по алгоритму №5:i=1: CPmax(Event\_1)=max{CP(Event\_1|Act\_1); CP(Even\_1|Act\_2)}= max{}=  
i=2: CPmax(Event\_2)=max{CP(Event\_2|Act\_1); CP(Even\_2|Act\_2)}= max{}=

Расчеты по алгоритму №6:i=1; j=1: COL(Event\_1|Act\_1)=CPmax(Event\_1) - CP(Event\_1|Act\_1)=  
i=1; j=2: COL(Event\_1|Act\_2)=CPmax (Event\_1) - CP(Event\_1|Act\_2)=  
i=2; j=1: COL(Event\_2|Act\_1)=CPmax (Event\_2) - CP(Event\_2|Act\_1)=   
i=2; j=2: COL(Event\_2|Act\_2)=CPmax (Event\_2) - CP(Event\_2|Act\_2)=

Расчеты по алгоритму №7:i=1; j=1: WOL(Event\_1|Act\_1)= COL(Event\_1|Act\_1)\*P(Event\_1)=  
i=1; j=2: WOL(Event\_1|Act\_2)= COL(Event\_1|Act\_2)\*P(Event\_1)=  
i=2; j=1: WOL(Event\_2|Act\_1)= COL(Event\_2|Act\_1)\*P(Event\_2)=  
i=2; j=2: WOL(Event\_2|Act\_2)= COL(Event\_2|Act\_2)\*P(Event\_2)=

Расчеты по алгоритму №8:j=1: EOL(Act\_1)=  
j=2: EOL(Act\_2)=

Расчеты по алгоритму №9:Min\_EOL=min{ EOL(Act\_1); EOL(Act\_2)}=min{}=

Расчеты по алгоритму №10:j=1: Так как EOL(Act\_1)= и, следовательно имеем \_=\_, то Solution=Act\_1 “Нажатие на кнопку Сильное вращение”;  
j=2: Так как EOL(Act\_2)= и следовательно \_≠\_, то Solution≠Act\_2.

## Результаты вычисления по алгоритмам 5-10

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Таблица №3: Переход от Условных прибылей к Условным возможным потерям | | | | | | |
| Событие | Вероятность | Act\_1 | Act\_2 | CPmax(Event\_i) | Act\_1 | Act\_2 |
| Условная прибыль ($) | | Условные возможные потери ($) | |
| Event\_1 | 0,3 | 800 | -2500 | 800 | 0 | 3300 |
| Event\_2 | 0,7 | 200 | 1000 | 1000 | 800 | 0 |
| Sum(Pj) | **1** | События являются взаимоисключающими и образуют полную группу! | | | | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Событие | Вероятность | Opt\_Act | CPmax | Act\_1 | Act\_2 | Act\_1 | Act\_2 |
| Условная возможные потери ($) | | Взвешенные возможные потери ($) | |
| Event\_1 | 0,3 | Act\_1 | 800 | 0 | 3300 | 0 | 990 |
| Event\_2 | 0,7 | Act\_2 | 1000 | 800 | 0 | 560 | 0 |
| Средние ожидаемые потери | | | | | | 560 | 990 |
| Sum(Pj)= | **1** | События являются взаимоисключающими и образуют полную группу! | | | | | |
| Решение | Нажать на кнопку **сильного** вращения | | | | | | |

## Рекомендация для клиента

Моему клиенту рекомендуется выбрать действие Act\_1 “**Нажать на кнопку Сильного вращения**”. Такое решение обеспечивает ему минимальное среднее ожидаемое значение потери равное 560 долларов. Он обеспечит себе такое минимальное среднее ожидаемое значение потери, если он ***бесконечно долго*** будет играть в эту игру, и при этом ***каждый раз*** будет нажимать на ***кнопку сильного вращения*** при заданных в Таблице 3. Исходных данных задачи. Это значит, что минимальное среднее ожидаемое значение потерь составит 560 долларов на одну игру.

# Матрица решений (для 2-х показателей – прибыль)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Событие\Действие | Вероятность | Нажать на кнопку **сильного** вращения | | Нажать на кнопку **слабого** вращения | |
| Условная премия ($) | Условный штраф ($) | Условная премия ($) | Условный штраф ($) |
| Колесо останавливается в **белой** области | 0,3 | 800 | 0 | 0 | 2500 |
| Колесо останавливается в **красной** области | 0,7 | 200 | 0 | 1000 | 0 |
| Sum(Pj) | **1** | События образуют полную группу! | | | |

Таблица 3. Матрица решений (Исходные данные задачи)

# Система обозначений для использования в описании алгоритмов решений задачи

**Примечание:** Из Таблицы №1: Исходные данные задачи видно, что даны только Премия и Штраф. Поэтому Условная Прибыль должна вычисляться по какой-то формуле.

**Prize(Event\_i | Act\_j)** это значение условной премии в данной комбинации события Event\_i и действия Act\_j, где i=1,n; j=1,m  
PRIZE(Event\_1 | Act\_1)= $800;  
PRIZE(Event\_1 | Act\_2)= $0;  
PRIZE(Event\_2 | Act\_1)= $200;  
PRIZE(Event\_2 | Act\_2)= $1000;

**Fine(Event\_i | Act\_j)** это значение условного штрафа в данной комбинации события Event\_i и действия Act\_j, где i=1,n; j=1,m

FINE(Event\_1 | Act\_1)= $0;  
FINE(Event\_1 | Act\_2)= $2500;  
FINE(Event\_2 | Act\_1)= $0;  
FINE(Event\_2 | Act\_2)= $0;

# Вычислительный алгоритм для решения задачи по критерию «Средняя ожидаемая прибыль»

Процесс решения проблемы по критерию «**Средняя ожидаемая прибыль**» состоит из вычислений по следующим 5 алгоритмам:

1. Алгоритм №1. Вычисление условной прибыли

## Описание процесса решения проблемы:

Расчеты по алгоритму №1:i=1; j=1: CP(Event\_1|Act\_1)= PRIZE(Event\_1 | Act\_1) – FINE(Event\_1 | Act\_1)=$800-$0=$800;  
i=1; j=2: CP(Event\_1|Act\_2)= PRIZE(Event\_1 | Act\_2) – FINE(Event\_1 | Act\_2)=$0—$2500= —$2500;  
i=2; j=1: CP(Event\_2|Act\_1)= PRIZE(Event\_2 | Act\_1) – FINE(Event\_2 | Act\_1)=$200-$0=$200;  
i=2; j=2: CP(Event\_2|Act\_2)= PRIZE(Event\_2 | Act\_2) – FINE(Event\_2 | Act\_2)=$1000-$0=$1000;