Lecture 4: Truth functions, evaluating compound statements

1. Functions, arithmetic functions, and truth functions

2. Definitions of truth functions

3. Evaluating compound expressions

4. Categorizing statements

Functions

A function is something that takes inputs and produces outputs.

You can think of them as a sort of *abstract* machine - like a bread machine that will produce as 'output' bread, if it is given as 'input' flour, yeast, sugar, etc.

Arithmetic Functions

Familiar examples of functions are the *arithmetic* functions addition, subtraction, multiplication and division

Addition takes 2 numbers as input, and produces 1 number as output

If you input 4 and 3, it outputs 7

Addition takes 2 numbers as input, and produces 1 number as output

If you input 4 and 3, it outputs 7

A function can be defined in terms of its entire input-output structure

x	y	x+y
1	1	2
1	2	3
2	1	3
2	2	4

A function can be defined in terms of its entire input-output structure

Addition '+'

x	y	x+y
1	1	2
1	2	3
2	1	3
2	2	4

Rickification '∧'

x	y	x∧y
1	1	2
1	2	3
2	1	3
2	2	4

Addition '+'

x	y	x+y
1	1	2
1	2	3
2	1	3
2	2	4

Subtraction '-'

x	y	x-y
1	1	0
1	2	-1
2	1	1
2	2	0

Multiplication 'x'

x	y	x x y
1	1	1
1	2	2
2	1	2
2	2	4

Division '+'

x	y	x÷y
1	1	1
1	2	.5
2	1	2
2	2	1

Truth Functions

Truth functions are functions that take truth values as inputs, and produce truth values as outputs.

There are two truth values:

True (T)

False (F)

Every simple statement is either True or False

The statement operators that form compound statements (conjunction, negation, etc.) symbolize *truth functions*.

H = I left you my house.
C = I left you my car.

I left you my house and I left you my car. (H • C)

H	C	H • C
T	T	T
T	F	F
F	T	F
F	F	F

Conjunction

φ	ψ	φ•ψ
T	T	T
T	F	F
F	T	F
F	F	F

Negation

I am married.

I am not married.

φ	~φ
T	F
F	T

Disjunction

Shelly won Lotto

Shelly got a big inheritance

Either Shelly won Lotto, or Shelly got a big inheritance.

Shelly either won Lotto or got a big inheritance

Disjunction

Shelly won Lotto

Shelly got a big inheritance

Shelly either won Lotto or got a big inheritance.

φ	ψ	φ v ψ
T	T	T
T	F	T
F	T	T
F	F	F

Disjunction

The expression 'or' in English is ambiguous, can express *two different* truth functions.

You can have soup or salad.

φ	ψ	φ v ψ
T	T	F
T	F	T
F	T	T
F	F	F

Disjunction

Inclusive OR

φ	ψ	φvψ
T	T	T
T	F	T
F	T	T
F	F	F

Exclusive OR

φ	ψ	φvψ
T	T	F
T	F	T
F	T	T
F	F	F

You can have soup or salad.

(P v D) • ~(P • D)

Conditional

You turn in all the homework.

I give you an A in the class.

If you turn in all the homework, I'll give you an A in the class.

φ	ψ	φ ⊃ ψ
T	T	T
T	F	F
F	T	T
F	F	T

Biconditional

You earn a C- or better.

I'll give you a P

I'll give you a P if and only if you earn a C- or better.

φ	ψ	φ ≡ ψ
T	T	T
T	F	F
F	T	F
F	F	T

Evaluating compound statements

a + b

a	b	a+b
1	1	2
1	2	3
2	1	3
2	2	4

Evaluating compound statements

(a x b) + (b - a)

a	b	a x b	b-a	(a x b) + (b-a)
1	1	1	0	1
1	2	2	1	3
2	1	2	-1	1
2	2	4	0	4

(P • Q) v (Q ≡ P)

Construct a Truth Table

Number of Rows = $2^n + 1$

Where *n* is the number of simple statements involved: $2^2 + 1 = 5$

(P • Q) v (Q ≡ P)

Construct a Truth Table

Number of Columns: 1 for each statement involved in the compound statement.

P; Q; P • Q; Q ≡ P; (P • Q) v (Q ≡ P) = 5

(P • Q) v (Q ≡ P)

P	Q			

In the first columns on the top row, put the simple statement in alphabetical order

(P • Q) v (Q ≡ P)

P	Q			
T	T			
T	F			
F	T			
F	F			

In the columns under the simple statements, fill out Ts and Fs so that every possible combination of truth values has a row

(P • Q) v (Q ≡ P)

P	Q			
T	T			
T	F			
F	T			
F	F			

In the first column, top half Ts and bottom half Fs.
For each subsequent column, alternate groups of Ts and Fs half the size of the groups of the previous column, until finished.

(P • Q) v (Q ≡ P)

P	Q	P • Q	Q ≡ P	(P • Q) v (Q ≡ P)
T	T			
T	F			
F	T			
F	F			

In subsequent columns, build up from the smallest compound statements to the largest, and fill in its truth column according to the truth function of that operator, and the truth values of the components

(P • Q) v (Q ≡ P)

P	Q	P • Q	Q ≡ P	(P • Q) v (Q ≡ P)
T	T	T		
T	F	F		
F	T	F		
F	F	F		

φ	ψ	φ•ψ
T	T	T
T	F	F
F	T	F
F	F	F

(P • Q) v (Q ≡ P)

P	Q	P • Q	Q ≡ P	(P • Q) v (Q ≡ P)
T	T	T	T	
T	F	F	F	
F	T	F	F	
F	F	F	T	

φ	ψ	φ≡ψ
T	T	T
T	F	F
F	T	F
F	F	T

(P • Q) v (Q ≡ P)

P	Q	P • Q	Q ≡ P	(P • Q) v (Q ≡ P)
T	T	T	T	T
T	F	F	F	F
F	T	F	F	F
F	F	F	T	T

φ	ψ	φvψ
T	T	T
T	F	T
F	T	T
F	F	F

~(~C ≡ D) ⊃ ~D

C	D	~C	~D	~C≡D	~(~C≡D)	~(~C≡D)⊃~D
T	T	F	F	F	T	F
T	F	F	T	T	F	T
F	T	T	F	T	F	T
F	F	T	T	F	T	T

~R ⊃ R

R	**~R**	**~R ⊃ R**
T	F	T
F	T	F

φ	ψ	φ⊃ψ
T	T	T
T	F	F
F	T	T
F	F	T

{(G v E) ⊃ ~K} ⊃ ~(~K • ~E)

E	G	K	~E	~K	GvE	(GvE) ⊃~K	~K● ~E	~(~K● ~E)	{(GvE)⊃~K} ⊃~(~K●~E)
T	T	T	F	F	T	F	F	T	T
T	T	F	F	T	T	T	F	T	T
T	F	T	F	F	T	F	F	T	T
T	F	F	F	T	T	T	F	T	T
F	T	T	T	F	T	F	F	T	T
F	T	F	T	T	T	T	T	F	F
F	F	T	T	F	F	T	F	T	T
F	F	F	T	T	F	T	T	F	F

Categorizing statements

It is often useful to know the range of truth values that a statement can have.

Statements that are always true are *tautologies*.

Statements that are always false are *contradictions*.

Statements that are can be either true or false are *contingencies*.

Categorizing statements

What does it mean to say that a statement might always be true? or always be false?

Arithmetic example: while n might be any number, we can know, because of the details of the multiplication function, that 2n will always be even.

n^2 will always be greater than or equal to zero.

Categorizing statements

By definition, simple statements are contingencies: they can be either true or false. This does not mean that all compound statements are contingencies, though.

P v ~P

P	~P	P v ~P
T	F	T
F	T	T

To determine whether a statement is a tautology, contradiction, or contingency, do a truth table for that statement, and look at its truth column:

- if it has only Ts, then it is a tautology
- if it has only Fs, then it is a contradiction
- if it has both Ts and Fs, then it is a contingency

P v ~P

P	~P	P v ~P
T	F	T
F	T	T

B • ~B

B	~B	B • ~B
T	F	F
F	T	F

R • ~(S v R)

R	S	S v R	~(S v R)	R • ~(S v R)
T	T	T	F	F
T	F	T	F	F
F	T	T	F	F
F	F	F	T	F