Lessons touched by this meeting according to schedule:

* 10. 11/11/2024
  + The diagonal method [§4.3]
* 11. 12/11/2024
  + The smn theorem (or parametrization theorem) [§4.4]
  + Universal function (some ideas)

Immagine che contiene testo, schermata, numero, Carattere

Descrizione generata automaticamenteImmagine che contiene testo, calligrafia, Carattere, bianco

Descrizione generata automaticamenteImmagine che contiene testo, schermata, Carattere, documento

Descrizione generata automaticamente

Immagine che contiene testo, schermata, Carattere, documento

Descrizione generata automaticamente

Immagine che contiene testo, Carattere, algebra

Descrizione generata automaticamenteHilbert hotel parallel: [here](https://www.reddit.com/r/learnmath/comments/nkujlh/cantors_diagonal_argument_vs_hilberts_hotel/) – (useful)

Suppose there exists a non-computable function f : N → N with the property that for any non-computable function g : N → N, the function f + g defined by (f + g)(x) = f(x) + g(x) is computable.

Since the quantification over g is universal, this property must also hold when g = f. That is, taking g = f, we would have that f + f = 2f is computable.

However, if 2f is computable, then f itself would be computable since:

f(x) = (2f(x))/2

This is a computable operation (division by 2) applied to a computable function (2f), which would make f computable.

This contradicts our initial assumption that f is non-computable.

Therefore, we conclude that no such non-computable function f can exist. □

Immagine che contiene testo, Carattere, schermata, linea

Descrizione generata automaticamenteThe key insight here is that if f has the property that its sum with any non-computable function is computable, then in particular it must have this property when added to itself. But this leads to a contradiction since it would make f computable.

The function f is not computable. We can prove this by showing that if f were computable, then K = {x | φx(x)↓} would be recursive.

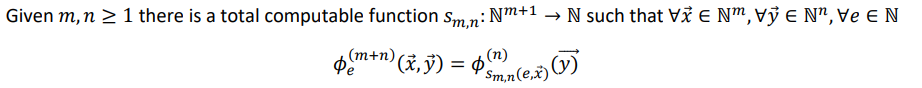
Suppose f is computable. Then we can write: χK(x) = sg(f(x) - (x+2))

This is because:

* If x ∈ K, then φx(x)↓, so f(x) = x+2, thus χK(x) = sg(0) = 0
* If x ∉ K, then φx(x)↑, so f(x) = x-1, thus χK(x) = sg(-(3)) = 1

Since this would express χK as a composition of computable functions (sg and f), it would make K recursive. However, we know that K is not recursive.

Therefore, f cannot be computable. □

Kleene’s smn-theorem

The SMN theorem states that given a function g(x, y) which is computable, there exists a total and computable function s such that phi\_s(x) (y) = g(x, y), basically "fixing" the first argument of g. It's like partially applying an argument to a function.

Usually you use it to create a reduction function by first finding an appropriate g(x, y) and then using the SMN theorem to say that there exists the previously cited function s which is also the reduction function. The difficult part is finding the appropriate function g(x, y), then the application of the SMN theorem is always the same.

Immagine che contiene testo, schermata, Carattere, informazione

Descrizione generata automaticamente

Immagine che contiene testo, Carattere, schermata, algebra

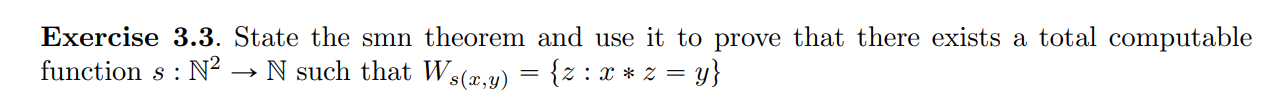
Descrizione generata automaticamenteThe proof works by showing how to construct a new URM program P' from an original program Pe that computes φe(m+n).

Immagine che contiene testo, Carattere, schermata, bianco

Descrizione generata automaticamenteImmagine che contiene testo, Carattere, schermata

Descrizione generata automaticamenteImmagine che contiene testo, schermata, Carattere, design

Descrizione generata automaticamente



1) First, let's state the smn theorem:

For any m,n ≥ 1, there exists a total computable function s(m,n): Nm+1 → N such that for all e ∈ N, x̄ ∈ Nm, ȳ ∈ Nn:

φ(m+n)e(x̄,ȳ) = φ(n)s(m,n)(e,x̄)(ȳ)

2) To solve our problem, let's first define a helper function f: N3 → N:

f(x,y,z) = { 0 if x \* z = y

↑ otherwise }

or equivalently:

f(x,y,z) = μw.|x\*z - y|

3) This function f is computable since:

- Multiplication is computable

- Absolute value is computable

- Bounded minimization of computable functions is computable

4) By the smn theorem (using m=2, n=1), there exists a total computable function s: N2 → N such that: For all x,y,z ∈ N: φs(x,y)(z) = f(x,y,z)

5) Let's verify that Ws(x,y) = {z : x \* z = y}:

- z ∈ Ws(x,y) ⟺ φs(x,y)(z)↓ ⟺ f(x,y,z)↓ ⟺ x \* z = y

Immagine che contiene testo, schermata, Carattere, documento

Descrizione generata automaticamente Therefore, s is the required function. □

Inside of the tutoring, the notation with was used, hence , which is the usual notation we have.

Why this choice of the helper function?

* When x divides n, we put x\*n in the output set
* Immagine che contiene testo, schermata, Carattere, documento

  Descrizione generata automaticamenteWhen x doesn't divide n, we put 1 (ensuring function is total)

Why this design of the helper function?

* When x ≥ n: outputs 2\*(x-n), which:
  + Controls domain through x ≥ n condition
  + Ensures even outputs by multiplying by 2
* When x < n: undefined (↑)

We then make the function computable - this works because:

* μz.(n-x) will be defined only when x ≥ n
* 2\*(x-n) ensures outputs are even

Immagine che contiene testo, schermata, Carattere, numero

Descrizione generata automaticamenteImmagine che contiene testo, schermata, Carattere, documento

Descrizione generata automaticamenteConcrete example on how we use the smn-theorem (this we’ll be explored in detail when it will be relevant):