VII. Imperative Real Programming

KAIST
CS700 M. Ziegler

- WHILE Programs:
  - over ℕ, over ℝ, over ℤ ℝ 𝕂=Kleene
- *Non*-Extensional Integer Rounding:
  - *naïve* and *fast*
- Exponential Function
- *Tri*section for Simple Unique Root Finding
- Matrix determinant with *multival.* pivoting
- ℤℝ𝕂 WHILE Programs: Turing-complete

Computing $f:\subseteq\mathbb{R}\to\mathbb{R}$ means: Convert any $\underline{a}=(a_m)\subseteq\mathbb{Z}$ with $|x-a_m/2^m|\leq 2^{-m}$, to $(b_n)\in\mathbb{Z}$ with $|f(x)-b_n/2^n|\leq 2^{-n}$

WHILE+ Program over ℕ

$\mathcal{P} := (x_j := 0 \mid x_j := 1 \mid x_j := x_i + x_k \mid x_j := x_i \ominus x_k$
$\mid x_j := x_i \oslash 2 \mid \mathcal{P} ; \mathcal{P} \mid \text{WHILE } x_j \text{ DO } \mathcal{P} \text{ END})$

loop executed as long as $x_j > 0$

$x > y \Leftrightarrow (x \ominus y) > 0$

IF $x>0$ THEN $\mathcal{P}$ ENDIF

$\Leftrightarrow$ ($t:=x$; WHILE t DO $\mathcal{P}$; $t:=0$; ENDWHILE)

$x=0$ $\Leftrightarrow$

($r:=0$; IF $x>0$ THEN $r:=0$; ENDIF ; "RETURN r")

FOR $N:=K$ TO L DO $\mathcal{P}$ ENDFOR $\Leftrightarrow$

$N:=K$; WHILE $(N<L+1)$ DO $\mathcal{P}$; $N:=N+1$; ENDWHILE

WHILE+ Program over $\mathbb{R}$

$\mathcal{P} := (\, x_j := 0 \mid x_j := 1 \mid x_j := x_i + x_k \mid x_j := x_i - x_k$
$\mid x_j := x_i \div 2 \mid \mathcal{P} ; \mathcal{P} \mid \text{WHILE } x_j \text{ DO } \mathcal{P} \text{ END} \,)$

loop executed as long as $x_j > 0$

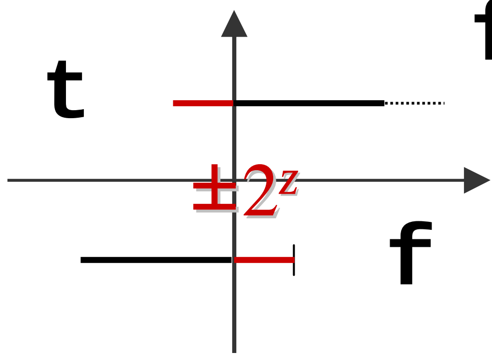
undecidable!

- *real* comparison "$x>0$" has *partial* semantics:
  - **t** for $x>0$, **f** for $x<0$, **u** for $x=0$: *Kleene* Logic $\mathbb{K}$
  - **u** different from *mathemat. undefined* $\uparrow = 1/0$
- *soft* comparison "$x >_z 0$" *multi*valued:

t for $x \geq -2^z$

f for $x \leq +2^z$

- integer rounding
  $\mathbb{R} \ni x \to \lfloor x \rfloor, \lceil x \rceil, \lfloor x \rceil \in \mathbb{Z}$
  *un*computable



$z \longrightarrow -\infty$

WHILE Program over $\mathbb{Z}\ \mathbb{R}\ \mathbb{K}$

KAIST
CS700 M. Ziegler

Types INTEGER=$\mathbb{Z}$, REAL=$\mathbb{R}$,

KLEENE=$\mathbb{K}$={**t**,**f**,**u**}

"Error" embedding
$\iota:\mathbb{Z}\ni z\to 2^z\in\mathbb{R}$

NOT(A)

A	¬A
F	T
U	U
T	F

AND(A, B)

A ∧ B		B		
		F	U	T
A	F	F	F	F
	U	F	U	U
	T	F	U	T

OR(A, B)

A ∨ B		B		
		F	U	T
A	F	F	U	T
	U	U	U	T
	T	T	T	T

- *real* comparison "$x>0$" has *partial* semantics:
  - **t** for $x>0$, **f** for $x<0$, **u** for $x=0$: *Kleene* Logic $\mathbb{K}$

choose$(k_1,\ldots,k_d)$ = *some* $j=1\ldots d$ s.t. $k_j=$ **t**

choose$(x<\iota(z)\ ,\ x>-\iota(z))$**-1 ==**

1 : $x\geq -\iota(z)$

0 : $x\leq +\iota(z)$

Theorem: The *three-sorted* structure

$(\mathbb{Z},0,1,+,-,\leq)\cup(\mathbb{R},0,1,+,-,\times,\leq)\cup(\mathbb{K},\mathbf{t},\mathbf{f},\mathbf{u},\neg,\vee,\wedge)$

with "*error*" embedding $\iota:\mathbb{Z}\ni z\to 2^z\in\mathbb{R}$ is decidable.

Naive Non-extensional Integer Rounding

Round: $\mathbb{R} \ni r \to z \in \mathbb{Z}$ s.t. $r\text{-}1 < z \leq r\text{+}1$ (**2** choices)

```
INTEGER Round (REAL r) {
INTEGER z:=0; REAL x:=r;   REAL zr:=0;
WHILE  choose( zr<r , z > r-1 ) == 1  DO
   z := z + 1;  zr:=zr+1;  END;
WHILE  choose( zr>r , z < r+1 ) == 1  DO
   z := z - 1;  zr:=zr-1;  END;
RETURN z; }
```

$(\mathbb{Z},0,1,+,>)$ and $(\mathbb{R},0,1,+,\times,>)$ with $\iota:\mathbb{Z}\ni p\to 2^p\in\mathbb{R}$

- *Partial* semantics of tests: „**x>y**" = **u** if $x=y$.
- **choose(x$_1$>y$_1$,…,x$_d$>y$_d$)** = *some j* s.t. $x_j>y_j$

Fast **Non-extensional Integer Rounding**

Round: $\mathbb{R} \ni r \to z \in \mathbb{Z}$ s.t. r-1 < $z \le r$+1 (**2** choices)

```
INTEGER Round (REAL r) {
INTEGER z:=0; REAL x:=r;  INTEGER b; INTEGER k:=0;
WHILE choose( |x|> ½ , |x| < 1 ) == 1 DO
  k:=k+1;   x:=x/2;  END;  // normalize to |x|<1
WHILE k>0 DO  x := x·2;
  b := choose( x<0 , -1<x<1 , x>0 ) - 2;
  // most signif. signed bin. digit -1,0,+1 of x
  x := x - b;  z := z + z + b;  k := k - 1;
  END; RETURN z; }
```

($\mathbb{Z}$,0,1,+,>) and ($\mathbb{R}$,0,1,+,×,>) with $\iota:\mathbb{Z}\ni p\to 2^p\in\mathbb{R}$

- *Partial* semantics of tests: „**x>y**" = **u** if x=y.
- **choose(x_1>y_1,…,x_d>y_d)** = *some j* s.t. $x_j > y_j$

Exponential Function

Lemma: For $x \leq 1$, $\left| \exp(x) - \sum_{j \leq -p} x^j / j! \right| \leq 2^p$

```
REAL Exp (INTEGER p; REAL x) {  // x ≤ 1
INTEGER j:=0; REAL rj:=0;        // j == rj
REAL sum:=0; REAL xj:=1;        // xj == x^j/j!
WHILE j≤-p DO
  sum := sum + xj;
  j := j+1;  rj := rj + 1;
  xj := xj × x / rj;
END;  RETURN sum; }
```

For arbitrary $x \in \mathbb{R}$, use
$\exp(x+y) = \exp(x) \cdot \exp(y)$

Arg.s given *exactly*, return *approx.* up to error 2^p

- *Partial* semantics of tests: „**x>y**“ = **u** if $x=y$.
- **choose(x$_1$>y$_1$,…,x$_d$>y$_d$)** = *some j* s.t. $x_j > y_j$

Recall Computability of `rootof`

Computable Intermediate Value Theorem:

Suppose $f:[0;1]\to[-1;1]$ is computable with $f(0)<0<f(1)$.
Then f has some computable root $x\in[0;1]$ with $f(x)=0$.

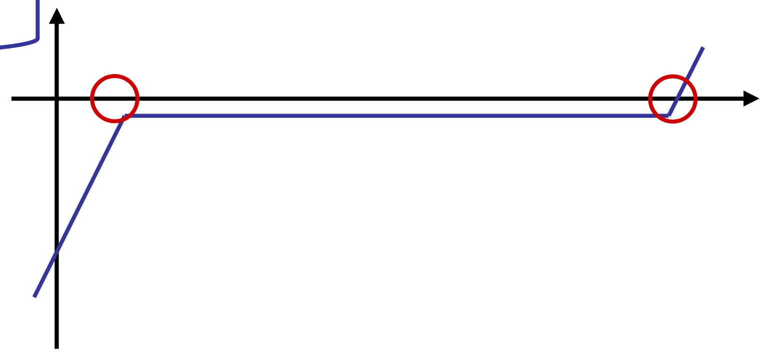
Proof (Bisection):

Initially $a:=0$, $b:=1$.

f has (at least one) root in $[a;b]$.

- If $f((a+b)/2)<0$ then let $a:=(a+b)/2$ and continue.
- If $f((a+b)/2)>0$ then let $b:=(a+b)/2$ and continue.
- If $f((a+b)/2)=0$ then return $(a+b)/2$.

n-th iteration: $|b-a| = 1/2^n$



f has a root in $\mathbb{D}$:

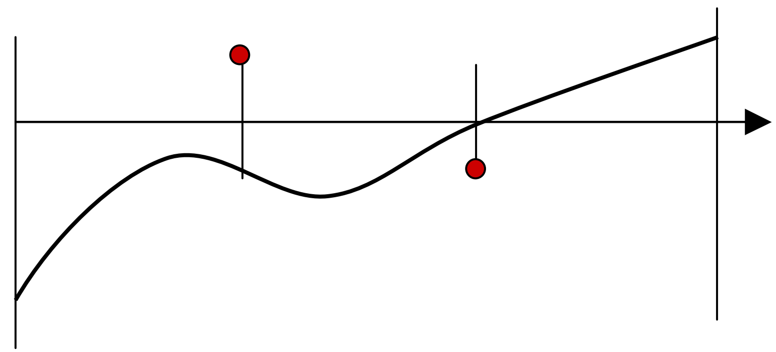
computable!

(f is fixed!)

"Any computable functional is continuous!"

*Tri*section for Simple Unique Root Finding


```
REAL Trisection (INTEGER p; [0;1]→REAL f) {
// f(x) < 0 < f(y),    ∃! z∈[0,1]: f(z)=0
REAL x:=0; REAL y:=1;
WHILE choose( y-x > ι(p-1), ι(p) > y-x ) == 1  DO
  IF choose ( 0>f((2x+y)/3) , 0<f((x+2y)/3) ) == 1
   THEN  x:=(2x+y)/3;
   ELSE  y:=(x+2y)/3;
   END;
END;  RETURN x; }
```



Arg.s given *exactly*, return *approx.* up to error 2^{p}

- *Partial* semantics of tests: „**x>y**" = **u** if $x=y$.
- **choose(**$x_1>y_1,\dots,x_d>y_d$**)** = *some j* s.t. $x_j>y_j$

Matrix Determinant Computation


```
REAL Det (INTEGER p; REAL M[d×d]) {    // M regular!
INTEGER i:=0,j:=0,k:=0;                 // full pivoting
INTEGER pi:=0,pj:=0; REAL det:=1;  // row  echelon:
FOR k:=0 TO d-2 DO                    // M[k..d-1,k..d-1]
 (pi,pj) := pivot(M,k);    // pi,pj s.t. M[pi,pj]≠0
 det := det × M[pi,pj];
 FOR j:=0 TO d-1 DO  swap(M[k,j],M[pi,j]);
 IF k≠pi THEN det := - det;
 FOR i:=0 TO d-1 DO  swap(M[i,k],M[i,pi]);
 IF k≠pj THEN det := - det;
 FOR j:=k+1 TO d-1       // scale row #k by 1/M[k,k]
   M[k,j] := M[k,j] / M[k,k];        // and subtract
   FOR i:=k+1 TO d-1 DO    // M[i,k]-fold from rows
     M[i,j] := M[i,j] – M[i,k] × M[k,j] //#k+1..d-1
   M[k,k] := 1; FOR i:=k+1 TO d-1 DO M[i,k]:=0;
 ENDFOR k;  RETURN det × M[d-1,d-1]; }
```

Multivalued Pivot Search


```
REAL Det (INTEGER p; REAL M[d×d]) {    // M regular!
INTEGER i:=0,j:=0,k:=0;                 // full pivoting
INTEGER pi:=0,pj:=0; REAL det:=1;   // row  echelon:
FOR k:=0 TO d-2 DO                     // M[k..d-1,k..d-1]
 (pi,pj) := pivot(M,k);     // pi,pj s.t. M[pi,pj]≠0
  pi:=k; pj:=k; REAL pv:=0;
  FOR i:=k TO d-1 DO
    FOR j:=k TO d-1 DO
      pv := max(pv,|M[i,j]|);
  FOR i:=k TO d-1 DO
    FOR j:=k TO d-1 DO
      IF choose( |M[i,j]|<pv , |M[i,j]|>pv/2 ) == 2
        THEN pi:=i;  pj:=j;
```

WHILE Program over $\mathbb{Z}\ \mathbb{R}\ \mathbb{K}$

Def: A $\mathbb{Z}\mathbb{R}\mathbb{K}$-WHILE Program for $f :\subseteq \mathbb{R}^d \times \mathbb{Z}^e \to \mathbb{R}$

- Receives e **INTEGER** and d **REAL** arguments exactly and one dedicated integer *precision* parameter p:

```
REAL F (INTEGER p; REAL r1,…,REAL rd, INTEGER z1,
        …,INTEGER ze)
```

- Employs a constant number of local variables of types **INTEGER**, **REAL**, **KLEENE**.
- Performs arithmetic $+,-,\times,\div$ on real data, exactly; with partial test ***x<y*** $\in$ **KLEENE**; multivalued **choose**
- Returns some real approximation to $f(r_1,..,r_d,z_1,..,z_e)$ up to absolute error $\iota(p)$, if $(r_1,\ldots,r_d,z_1,\ldots,z_e) \in \mathrm{dom}(f)$

$(\mathbb{Z},0,1,+,-,\leq) \cup (\mathbb{R},0,1,+,-,\times,\leq) \cup (\mathbb{K},\mathbf{t},\mathbf{f},\mathbf{u},\neg,\vee,\wedge)$

3-sorted structure with embedding $\iota : \mathbb{Z} \ni z \to 2^z \in \mathbb{R}$

$\mathbb{ZRK}$-WHILE is Turing-complete

Def: A $\mathbb{ZRK}$-WHILE Program for $f{:}\subseteq\mathbb{R}^d\times\mathbb{Z}^e\to\mathbb{R}$

Theorem: $\mathbb{ZRK}$-WHILE Programs are "*Turing-complete for the reals*": They can express • any and • only *computable* partial $f{:}\subseteq\mathbb{R}^d\times\mathbb{Z}^e\to\mathbb{R}$.

Proof (Sketch) $\Leftarrow$**:** Real arithmetic is computable, and so are *partial* real tests and *multi*valued **choose**. Limits $p\to-\infty$ of approx.s to error 2^p are computable.

Proof (Sketch) $\Rightarrow$**:** Given $x\in\mathrm{dom}(f)$, let $a_m{:}=\mathrm{Round}(x\cdot 2^m)$.

Round: $\mathbb{R}\ni r\to z\in\mathbb{Z}$ s.t. $r\text{-}1<z\leq r\text{+}1$ (**2** choices)

Computing $f{:}\subseteq\mathbb{R}\to\mathbb{R}$ means: Convert any $\underline{a}=(a_m)\subseteq\mathbb{Z}$ with $|x\text{-}a_m/2^m|\leq 2^{-m}$, to $(b_n)\in\mathbb{Z}$ with $|f(x)\text{-}b_n/2^n|\leq 2^{-n}$

VII. Imperative Real Programming

- WHILE Programs:
  - over $\mathbb{N}$, over $\mathbb{R}$, over $\mathbb{Z}\ \mathbb{R}\ \mathbb{K}$=Kleene
- *Non*-Extensional Integer Rounding:
  - *naïve* and *fast*
- Exponential Function
- *Tri*section for Simple Unique Root Finding
- Matrix determinant with *multival.* pivoting
- $\mathbb{Z}\mathbb{R}\mathbb{K}$ WHILE Programs: Turing-complete