OEP
2.gyakorlat
Egyszerű reprezentációjú típusok megvalósítása

1. Feladat – racionális számok

Valósítsuk meg a racionális számok típusát úgy, hogy kihasználjuk azt, hogy minden racionális szám ábrázolható két egész számmal, mint azok hányadosa! Implementáljuk az alapműveleteket!

Típusdefiníció

$\mathbb{Q}$	c := a±b (a:$\mathbb{Q}$, b:$\mathbb{Q}$, c:$\mathbb{Q}$)
	c := a*b (a:$\mathbb{Q}$, b:$\mathbb{Q}$, c:$\mathbb{Q}$)
	c := a/b (b≠0) (a:$\mathbb{Q}$, b:$\mathbb{Q}$, c:$\mathbb{Q}$)
x,y: $\mathbb{Z}$ (Inv: y≠0) // $\frac{x}{y}$	c.x, c.y := a.x*b.y ± a.y*b.x, a.y*b.y
	c.x, c.y := a.x * b.x, a.y * b.y
	c.x, c.y := a.x * b.y, a.y * b.x (b.x≠0)

2. Feladat – komplex számok

Valósítsuk meg a komplex számok típusát! Ábrázoljuk a komplex számokat az algebrai alakjukkal (x+iy)! Implementáljuk az alapműveleteket!

Típusdefiníció:

$\mathbb{C}$	c := a±b (a: $\mathbb{C}$, b:$\mathbb{C}$, c:$\mathbb{C}$)
	c := a*b (a:$\mathbb{C}$, b:$\mathbb{C}$, c:$\mathbb{C}$)
	c := a/b (a:$\mathbb{C}$, b:$\mathbb{C}$, c:$\mathbb{C}$) b≠0
x,y: $\mathbb{R}$ // x+iy	c.x, c.y := a.x ± b.x, a.y ± b.y
	c.x, c.y := a.x * b.x – a.y * b.y, a.x * b.y + a.y * b.x
	c.x, c.y := (a.x * b.x + a.y * b.y)/(b.x^2+ b.y^2), (a.y * b.x – a.x * b.y)/(b.x^2+ b.y^2) b.x≠0 ∨ b.y≠0

3. Pont, Kör

Valósítsuk meg a pont típus, mely a síkbeli koordináta-rendszer pontjainak a kezelésére lesz alkalmas! Implementáljuk két pont távolságát kiszámoló metódust!

Pont	d:= távolság(p, q) (p, q :Pont, d:$\mathbb{R}$)
(x, y) :$\mathbb{R}^2$	d:= $\sqrt{(p.x - q.x)^2 + (p.y - q.y)^2}$

Point
+ x : real + y : real
+ distance(Point) : real

Valósítsunk meg egy kör típust, amely használja a pont típust! Implementáljuk annak eldöntését, hogy egy adott pont rajta van-e a körlemezen.

Kör	l := tartalmaz(k, p) (k:Kör, p:Pont, l:$\mathbb{L}$)
c : Pont, r : $\mathbb{R}$ Inv: r ≥ 0	l := távolság(k.c, p) ≤ k.r

Circle
- c : Point - r: real { r ≥ 0 }
+ contains(p : Point) : bool <<getter>> + getc() : Point + getr() : real

Feladat a típus használatára:

Adott egy síkbeli pontokat tartalmazó tömb, és egy kör. A tömbnek van-e olyan eleme, amely a körben helyezkedik el? Ha igen, add meg az első ilyen pontot!

Specifikáció:

A = (x:Point^n, k:Circle, l:$\mathbb{L}$, ind : $\mathbb{Z}$)

Ef = (x=x' $\wedge$ k=k')

Uf =(Ef $\wedge$ (l, ind) = SEARCH$_{i=1..n}$ k.contains(x[i]))

Algoritmus:

i, l := 1, false	i:$\mathbb{Z}$
$\neg$l $\wedge$ i ≤ n	
l, ind := k.contains(x[i]), i	
i := i+1	

4. Síkvektorok típusa

Valósítsuk meg a síkvektorok típusát! Implementáljuk két vektor összegének, egy vektor nyújtásának, és két vektor skaláris szorzatának műveleteit!

Vektor	c := a+b	(a, b, c : Vektor)
	c := k*a	(a, c : Vektor, k:$\mathbb{R}$)
	s := a*b	(a, b: Vektor, s:$\mathbb{R}$)

„origóból az (x_1, x_2) koordinátájú pontba mutató vektorok"

x, y:$\mathbb{R}$	c.x, c.y := a.x+b.x, a.y+b.y
	c.x, c.y := k*a.x, k*a.y
	c.x, c.y:= a.x*b.x + a.y*b.y

<u>Feladat a típus használatára:</u>

Adott síkvektorok összege merőleges-e egy adott síkvektorra (skaláris szorzatuk nulla-e).

Specifikáció:

A = (v:Vektor ^{n}, sv:Vektor, l: $\mathbb{L}$)

Ef = (v=v' $\wedge$ sv=sv')

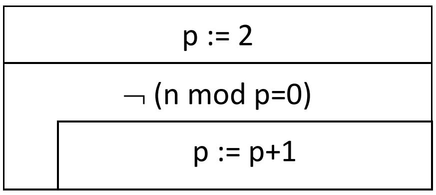
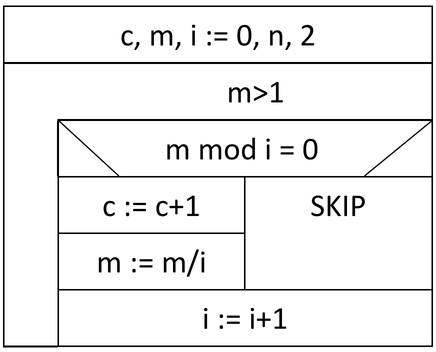
Uf = (Ef $\wedge$ l = (($\sum_{i=1..n}$ v[i])*sv = 0.0))

Algoritmus:

s := 0	s:Vektor
i = 1..n	i:$\mathbb{Z}$
s := s + v[i]	
l := (s*sv=0)	

5. Prímek halmaza

Valósítsuk meg a prímek halmazának típusát a következőképpen: A prímek halmazát egy természetes számmal tudjuk reprezentálni úgy, hogy a szám a halmazban szereplő prímek szorzata. Implementáljuk a következő műveleteket: benne van-e, betesz, kivesz, üres, üres-e, minimum, elemszám.

set($\mathbb{P}$) prímek véges halmaza	l := h.benne_van-e (p) (h : set($\mathbb{P}$), p : $\mathbb{P}$, l : $\mathbb{L}$)
	h.betesz(p) (h : set($\mathbb{P}$), p : $\mathbb{P}$) // ha p-t h már tartalmazza, akkor hatástalan
	h.kivesz(p) (h : set($\mathbb{P}$), p : $\mathbb{P}$) // ha p-t h nem tartalmazza, akkor hatástalan
	h.üres() (h : set($\mathbb{P}$))
	l := h.üres-e() (h : set($\mathbb{P}$), l : $\mathbb{L}$)
	p := h.min() (h : set($\mathbb{P}$), p : Prím) // hiba, ha p üres halmaz
	c := h.elemszám() (h : set($\mathbb{P}$), c : $\mathbb{N}$)
n : $\mathbb{N}$ ahol az n szám prímtényezős felbontásában a prímek egyszeresen fordulnak elő	l := (n mod p=0)
	n := n· p ha n mod p $\neq$ 0
	n := n/p ha n mod p = 0
	n := 1
	l := (n=1)
	 kiválasztás
	 m, i : $\mathbb{Z}$ számlálás

6. Másodfokú polinomok

Hozzunk létre a másodfokú polinomok típusát. Implementáljuk a következő műveleteket:

- Helyettesítési értékének kiszámítása
- Polinomok összeadása, szorzása számmal

<table>
<tr><td rowspan="3">másodfokú polinomok (MP)</td><td>h := érték(p, x)
(p : MP, x : $\mathbb{R}$, h : $\mathbb{R}$)</td></tr>
<tr><td>p3 := p1±p2 (p1, p2, p3:MP)</td></tr>
<tr><td>p2 := p1*q (p1, p2:MP, q: $\mathbb{R}$)</td></tr>
<tr><td rowspan="3">a,b,c : $\mathbb{R}$
// ax^2+bx+c</td><td>h := p1.a*x*x+p1.b*x+p1.c</td></tr>
<tr><td>p3.a, p3.b, p3.c :=
p1.a ± p2.a, p1.b ± p2.b, p1.c ± p2.c</td></tr>
<tr><td>p3.a, p3.b, p3.c :=
p3.a*q, p3.b*q, p3.c*q</td></tr>
</table>

Polinom (MP)
- a : real - b : real - c : real
+ value(x : real) : real + add(p1 : MP, p2 : MP) : MP + mul(p1 : MP, q : real) : MP

<u>Feladat a típus használatára:</u>

Adott egy valós számokat tartalmazó vektor, és egy másodfokú polinom. A vektor mely elemére ad a másodfokú polinom maximális helyettesítési értéket?

Specifikáció:

A = (p : MP, v : $\mathbb{R}^n$, ind : $\mathbb{Z}$, max : $\mathbb{R}$)

Ef = (p=p' $\wedge$ v=v')

Uf = (Ef $\wedge$ (max, ind) = $\mathrm{MAX}_{i=1..n}$ érték(p, v[i]))

Algoritmus:

<table>
<tr><td colspan="3">max, ind, := érték(p, v[1]),1</td><td>$i:\mathbb{Z}$</td></tr>
<tr><td colspan="3">i = 2.. n</td></tr>
<tr><td rowspan="2"></td><td colspan="2">max < érték(p, v[i])</td></tr>
<tr><td>max, ind :=
érték(p, v[i]), i</td><td>SKIP</td></tr>
</table>