56

a.

Phân-Số ::

Tử-số: R

Mẫu-số: R

Inv-Phân-Số: Phân-Số B

Inv-Phân-Số (ps) let ms = ps.Mẫu-Số in ms 0

b.

Tổng-Phân-Số: Phân-Số\* Phân-Số

Tổng-Phân-Số (pss)

If (len(pss) 0)

Then 0

Else kq = Rút-Gọn-PS(Cộng-Phân-Số(head(pss) + Tổng-Phân-Số(tail(pss))))

58

a.

Điểm::

x: R

y: R

Đường-Thẳng::

d1: Điểm

d2: Điểm

Điểm-Thuộc-ĐT: Đường-Thẳng x Điểm B

Điểm-Thuộc-ĐT (dt,d) let

xAB = dt.d2.x – dt.d1.x,

yAB = dt.d2.y – dt.d1.y,

xAC = d.x – dt.d1.x,

yAC = d.y – dt.d1.y,

xBC = d.x – dt.d2.x,

yBC = d.y – dt.d2.y in if ((xAB / xBC) = (yAB / yBC)) then true else false

59.

Điểm::

x: R

y: R

Tam-Giác::

d1: Điểm

d2: Điểm

d3: Điểm

Inv-Tam-Giác: Tam-Giác B

Inv-Tam-Giác (tg)

(Điểm-Thuộc-ĐT(mk-Đường-Thẳng(tg.d1,tg.d2), d3)

(Điểm-Thuộc-ĐT(mk-Đường-Thẳng(tg.d2,tg.d3), d1)

(Điểm-Thuộc-ĐT(mk-Đường-Thẳng(tg.d3,tg.d1), d2)

60

a.

Stack::

mang: N\*

len: N

Inv-Stack: Stack B

Inv-Stack (s) (s.len 0)

b.

Stack-Push: Stack x N B

Stack-Push (s, e) let (ns = .mang.Thêm-Phần-Tử(e)) in Inv-Stack(mk-Stack(ns, .len))

Stack-Pop: Stack Z

Stack-Pop (s) let ns = .mang.Xóa-Phần-Tử-Tại(s.len) in (if (len(s.mang) 0) then -1 else let rs = head(reserve(s.mang)), s = mk-Stack(ns, .len) in rs)

Stack-Top: Stack Z

Stack-Top (s) if (len(s.mang) 0) then -1 else let rs = head(reserve(s.mang))

Stack-IsEmpty: Stack B

Stack-IsEmpty (s) if (len(s.mang) 0) then true else false

Stack-IsFull: Stack B

Stack-IsFull (s) if (len(s.mang) s.len) then true else false

Stack-Empty: Stack Stack

Stack- Empty (s) let ns = {} in mk-Stack(ns, .len)