Software Eng. 2AO4 - Software Design I - 2001
Laboratory 4

For weeks starting October 26 and November 2.

Laboratory 3 required 18 programs and dealt with just two rectangles. In this exercise, you are to repeat Laboratory 3 but implement your module so that you can store up to 100 different rectangles, each rectangle identified by an integer between 1 and 100 (inclusive). Initially there should be no rectangles stored.

Def: $ok(n) \equiv 0 < n \wedge n < 101$

1. A procedure, S2P, that takes 5 integer arguments: (n, x_0, y_0, x_1, y_1). The first parameter, n, which must be between 1 and 100, identifies one of 100 possible rectangles. The next pair of arguments gives the (x,y) coordinates of the upper left corner of rectangle n. This procedure will return a value as specified by table 1 using the following definitions.

Def: $isrectangle \equiv x_0 < x_1 \wedge y_0 > y_1$

	return value
$\neg$ ok(n)	-1
ok(n) $\wedge \neg$ isrectangle	-2
ok(n) $\wedge$ isrectangle	0

Table 1: Return value for S2P, SPHW, **and** SPHA

Additional effects of this procedure are defined in tables 2, 3, 4, 5, 6, 7, 8, and 9 below.

A procedure, SPHW, that takes 5 integer arguments: (n, x, y, h, w) representing a rectangle. The first parameter, n, which must be between 1 and 100, identifies one of 100 possible rectangles. The next pair of arguments gives the (x,y) coordinates of the upper left corner of the rectangle. The next pair gives the height and width of the rectangle. This procedure will return a value as specified by table 1 using the following definition. Additional effects of this procedure are defined in tables 2, 3, 4, 5, 6, 7, 8, and 9 below.

Def: $isrectangle \equiv h > 0 \wedge w > 0$

2. A procedure, SPHA, that takes 5 integer arguments: (n, x, y, h, s) representing a rectangle. The first parameter, n, which must be between 1 and 100, identifies one of 100 possible rectangles. The next pair of arguments gives the (x,y) coordinates of the upper left corner of the rectangle. The next pair gives the height and area of the rectangle. This procedure will return a value as specified by table 1 using the following definitions. Additional effects are defined in tables 2, 3,4,5,6,7,8, and 9.

Def: $natural(x) \equiv x > 0 \wedge integer(x)$

Def: $isrectangle \equiv h > 0 \wedge natural(s/h)$

3. START(n) changes rectangle n to its initial state, i.e. no data stored. Return 0 if there was data stored previous to the call, -1 otherwise. Additional effects of this procedure are defined in tables 2, 3, 4, 5, 6, 7, 8, and 9 below.

4. A procedure rectangle(n), where n should be an integer, that returns an boolean value as specified by the following table. Initially rectangle(n) returns **false** for all values of n.

Most recent call of an "S" procedure for n is	START(n)	S2P (n,x_0,y_0,x_1,y_1)	SPHW (n,x, y, h,w)	SPHA (n, x, y, h, s)
	false	$x_0 < x_1 \wedge y_0 > y_1$	$h > 0 \wedge w > 0$	$h > 0 \wedge$ natural(s/h)_

Table 2: Value returned by rectangle(n)

5. A procedure GA(n) that takes one parameter, n, where n should be an integer between 1 and 100 and returns an integer value specified by the following table

Most recent call of an "S" procedure for[a] n is	START(n)	S2P (n,x_0,y_0,x_1,y_1)	SPHW (n,x, y, h,w)	SPHA (n, x, y, h, s)
$\neg$ ok(n) $\vee \neg$ rectangle(n)	-1	-1	-1	-1
ok(n) $\wedge$ rectangle(n)	-2	$(x_1-x_0) \times (y_0-y_1)$	$h \times w$	s

Table 3: Values returned by GA

a. If the most recent S procedure call was START(n), isrectangle will always be ***false***. The definition of isrectangle is that given with each of the S procedures. **This note applies to tables 3 through 9.**

6. A procedure GH(n) that takes one parameter, n, where n should be an integer between 1 and 100 and returns an integer value specified by the following table

Most recent call of an "S" procedure for n is	START(n)	S2P (n,x_0,y_0,x_1,y_1)	SPHW (n,x, y, h,w)	SPHA (n, x, y, h, s)
$\neg$ ok(n) $\vee \neg$ rectangle(n)	-1	-1	-1	-1
ok(n) $\wedge$ rectangle(n)	-2	$y_0 - y_1$	h	h

Table 4: Values returned by GH

7. A procedure GW(n) that takes one parameter, n, where n should be an integer between 1 and 100 and returns an integer value specified by the following table

Most recent call of an "S" procedure for n is	START(n)	S2P (n,x_0,y_0,x_1,y_1)	SPHW (n,x, y, h,w)	SPHA (n, x, y, h, s)
$\neg$ ok(n) $\vee \neg$ rectangle(n)	-1	-1	-1	-1
ok(n) $\wedge$ rectangle(n)	-2	$x_1 - x_0$	w	s/h

Table 5: Value returned by GW

8. Two procedures GULψ(n) that takes one parameter, n, where n should be an integer between 1 and 100 and ψ may be either x or y, that return an integer value specified by the following table

Most recent call of an "S" procedure for n is	START(n)	S2P (n,x_0,y_0,x_1,y_1)	SPHW (n,x, y, h,w)	SPHA (n, x, y, h, s)
$\neg$ ok(n) $\vee \neg$ rectangle(n)	-1	-1	-1	-1
ok(n) $\wedge$ rectangle(n)	-2	ψ_0	ψ_0	ψ_0

Table 6: Value returned by GULx, GULy,

9. A procedure GLRx(n) that takes one parameter, n, where n should be an integer between 1 and 100 and returns an integer value specified by the following table

Most recent call of an "S" procedure for n is	START(n)	S2P (n,x_0,y_0,x_1,y_1)	SPHW (n,x, y, h,w)	SPHA (n, x, y, h, s)
$\neg$ ok(n) $\vee \neg$ rectangle(n)	-1	-1	-1	-1
ok(n) $\wedge$ rectangle(n)	-2	x_1	x + w	x+ s/h

Table 7: Values returned by GLRx

10. A procedure GLRy(n) that takes one parameter, n, where n should be an integer between 1 and 100 and returns an integer value specified by the following table

Most recent call of an "S" procedure for n is	START(n)	S2P (n,x_0,y_0,x_1,y_1)	SPHW (n,x, y, h,w)	SPHA (n, x, y, h, s)
$\neg$ ok(n) $\vee \neg$ rectangle(n)	-1	-1	-1	-1
ok(n) $\wedge$ rectangle(n)	-2	y_1	y-h	y-h

Table 8: Values returned by GLRy

11. A procedure, `overlap(A,B)`, where A and B should both be an integer between 1 and 100. overlap must return an integer return value as described by the table below**.**

Def: ABok ≡ ok(A) ∧ ok(B)

	AinB	BinA	disjoint	overlap	AisB	tangent
ABok ∧ ¬ rectangle(A) ∧ rectangle(B)	-1	-1	-1	-1	-1	-1
ABok ∧ ¬ rectangle(B) ∧ rectangle(A)	-2	-2	-2	-2	-2	-2
¬ ABok ∨ (ABok ∧ ¬(rectangle(A) ∨ rectangle(B)))	-3	-3	-3	-3	-3	-3
ABok ∧ rectangle(A) ∧ rectangle(B)	1	2	3	4	5	6

Table 9: Return value for procedure `overlap(A,B)`

The definitions of AinB, BinA, disjoint, overlap, AisB, and tangent are those used in Laboratory 2 with the substitution of

- GULψ(ξ) for $\psi_{\xi 0,}$ where ξ may be either A or B and ψ may be either x or y,
- GξLRx(ξ), for $x_{\xi 1}$, where ξ may be either A or B,
- GLRy(ξ), for $y_{\xi 1}$ where ξ may be either A or B.

Please note that *every* module must include ***all*** *o*f the programs. You will not be able to mix programs from different modules. A user should not be able to tell which program was used to enter the rectangle's dimensions after the call to the S program is complete.