

Solve the following problems:

- 1) Problem 10 in page 141, but after adding another case "e", where both parameters x and y are passed by *value-result*. For each case, show the changes to the values of all involved variables (i.e., array A and i, in the caller block; and x and y in the callee P) as you step executing in the code. (45 pts)

Change the array values to be:

A[1] = 10

A[2] = 5

A[3] = 15

10. What values are printed by the following Algol program if
- x is passed by value and y is passed by value;
  - x is passed by value and y is passed by name;
  - x is passed by name and y is passed by value; and
  - x is passed by name and y is passed by name?

```
begin integer i, j; integer array A[1:3]
  procedure P(x,y); integer x,y;
    begin
      y := 2;
      Print(x);
      i := 3;
      Print(x);
      i := 3;
      Print(x)
      Print(y)
    end;

  A[1] := 7;   A[2] := 11;   A[3] := 13;
  i := 1;
  P (A[i], i);
  P (i, A[i])
end
```

2) Given the following ALGOL code:

(55 pts)

(\* please notice any text between (\*.\*) is a comment. Also, all labels blk1, blk2, blk3 are not part of the program, they just to show you the starting of blocks' codes.\*)

**main:** **begin** (\* main program block definition \*)

integer h, m; real n;

procedure **P** (k); value k ; integer k;

**blk1:** **begin** integer o;

procedure **R**(q); value q; real q;

**blk2:** **begin** real v; v := q + m; PrintInteger(v); **end** (\* blk2\*);

procedure **S** (i); value i; integer i;

**blk3:** **begin** integer m; m := i + 15; **R**(m) **end** (\*end blk3\*);

(\* **blk1**—**P** code \*)

o := k + m; **S**(o);

**end** (\* **blk1**—**P** code \*)

**begin** (\* the **main** program block code \*)

h := 25; m := 30; n := 2.7; **P**(h)

**end** (\* end of **main** program (block) code \*)

**end** (\* end of **main** program (block) definition \*)

- a) What would be the printed value of "v", in **blk2** above, in each of the following *scoping* mechanisms: i) **static**? ii) **dynamic**?

You **must** show the picture of the system stack upon the execution of the "PrintInteger(v)".

(20 pts)

- b) Draw the contour diagram of the above ALGOL program and answer the following questions, based on that diagram: (6 pts)

i) Can we call **P**() in **blk2**?

ii) Can we call **S**() in the main block **main**?

iii) Can we call "**blk3**" in **R**()?

- c) To improve our code, we have created a sibling procedure to the **P**(), called **Q1**(), which is declared immediately after **P**(). **Q1**() will have an internal procedure **Q11**() that declares an internal procedure **Q111**(). Remember, both **P**() and **Q1**() are siblings in the "main" block (program).

Here the layout of the code explained in part "c", in the above paragraph:

procedure **P** (); (interior code here as shown above); end;

.....

procedure **Q1** ();

(\***Q1** is sibling of **P**\*)

begin procedure **Q11** ();

(\***Q11** is a child of **Q1**\*)

begin procedure **Q111** (); end;

(\***Q111** is a child of **Q11**\*)

end;

end;

.....

Now, answer the following questions:

i) List the names of all procedures that are visible to be called from within the code of: Q1(), Q111(), R(), and P(); assuming static scoping.

(12 pts)

ii) If we assume dynamic scoping, can we still use the contour diagram to answer part "i" above? **Justify** your yes/no answer.

(7 pts)

iii) In case of dynamic scoping, is there a scenario that allows the calling of procedure Q111() from within S()? **ONLY** when you answer **YES**, show such scenario.

(10 pts)

**Extra Credit: (10 pts)**

i) When designing a HLL with the goal of increasing its user's domain (popularity of use starting with high school to university students), which **scoping** mechanism to utilize in such newly designed HLL: the **dynamic** or **static scoping**? You **must** justify your answer.