INF3110/4110 Solution - Exam 2004

Oppgave 1 Runtime-systems 24 %

1a
No, a runtime check is required. Ra may at different points of execution refer either an A object or a B object.

1b
A runtime check will still be required. Virtual methods are not only called remotely. Consider the following example

```
class A {
      void p(int i,int j){};
      void q(){p(1,2);};
}
class B extends A {
      void p extendedBy(int k){};
};
class C {
      void m() {
            A ra;
            ra = new B();
            ra.q();
      };
}
```

The call ‘p(1,2)’ in q in A will not always be a call to p of A. If it is executed as part of a B object, then the B.p method will be executed.

Oppgave 2 Skop 24%

2a
static scoping: (3): j=1

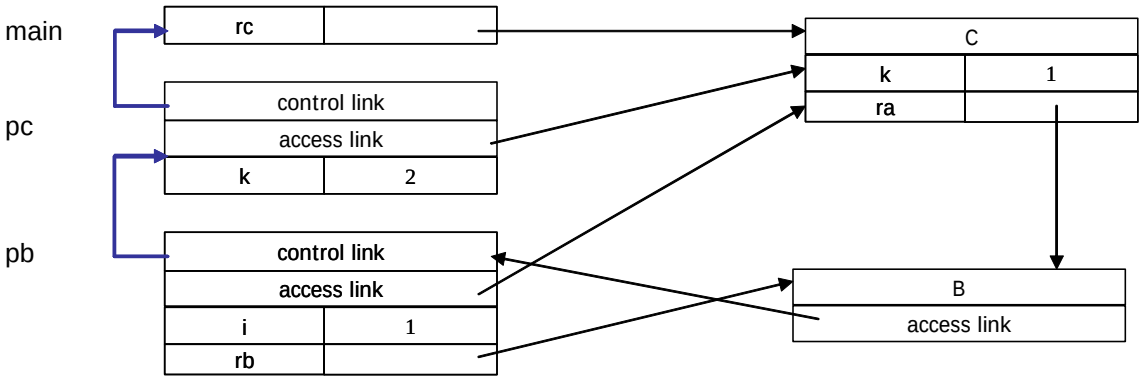
(4): j=3

dynamic scoping: (3): egentlig udefinert (da ‘i’ ikke er synlig), men j=0 eller j=2 aksepteres

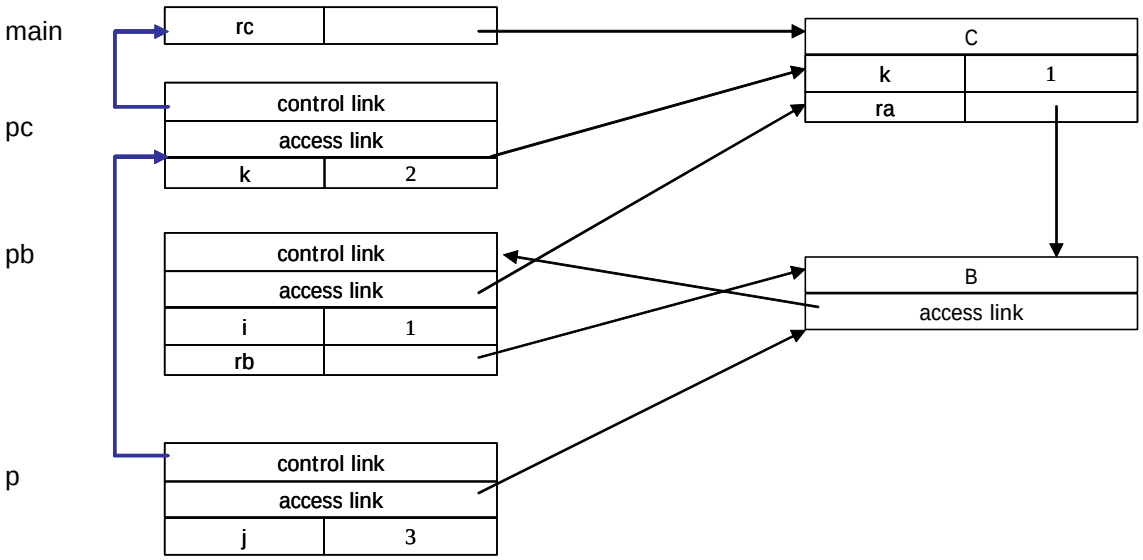
(4): j=3

2b

1)



2)



2c

No, the ‘pb’ activation record has to be maintained in order for ‘p’ to access the ‘i’ variable.

Oppgave 3 ML   30 %

3a

```
fun is_father (p1, MAN(_,fx,_)) = p1=fx
  | is_father (p1, WOMAN(_,fx,_)) = p1=fx
  | is_father (_, _) = false;
```

3b

```
fun is_child (MAN(mx,fx,_), p2) = mx=p2 orelse fx=p2
  | is_child (WOMAN(mx,fx,_), p2) = mx=p2 orelse fx=p2
  | is_child (_, _) = false;
```

3c

```
fun yob (MAN(_,_,y)) = y
  | yob (WOMAN(_,_,y)) = y;

fun oldest (p1, p2) = if yob(p1)<yob(p2) then p1 else p2;
```

3d

```
fun is_brother (MAN(m1,f1,_), MAN(m2,f2,_)) = m1=m2 orelse f1=f2
  | is_brother (MAN(m1,f1,_), WOMAN(m2,f2,_)) = m1=m2 orelse f1=f2
  | is_brother (_,_) = false;
fun is_sister (WOMAN(m1,f1,_), MAN(m2,f2,_)) = m1=m2 orelse f1=f2
  | is_sister (WOMAN(m1,f1,_), WOMAN(m2,f2,_)) = m1=m2 orelse f1=f2
  | is_sister (_,_) = false;

fun sibling (p1, p2) = is_brother(p1,p2) orelse is_sister(p1,p2);
```

3e

```
fun father_of (MAN(_,fx,_)) = fx
  | father_of (WOMAN(_,fx,_)) = fx;
fun mother_of (MAN(mx,_,_)) = mx
  | mother_of (WOMAN(mx,_,_)) = mx;

fun is_ancestor (NN, _) = false
  | is_ancestor (_, NN) = false
  | is_ancestor (p1,p2) =
        is_child(p2,p1) orelse
        is_ancestor(p1,father_of(p2)) orelse
      is_ancestor(p1,mother_of(p2));
```

Oppgave 4 Prolog   22 %

4a

```
grandchild(X,Y):- child(X,Z),child(Z,Y).
```

4b

```
father(X,Y):- child(Y,X),sex(X,male).
mother(X,Y):- child(Y,X),sex(X,female).


grandfather(X,Y):-father(Z,Y),father(X,Z).
grandfather(X,Y):-mother(Z,Y),father(X,Z).

grandmother(X,Y):-mother(Z,Y),mother(X,Z).
grandmother(X,Y):-father(Z,Y),mother(X,Z).
```

4c

```
inlaw(X,Y):- married(Y,Z),child(Z,X).

fatherinlaw(X,Y):- married(Y,Z),child(Z,X),sex(X,male).
```

4d

```
ancestor(X,Y):- child(Y,X).
ancestor(X,Y):- child(Y,Z),ancestor(Z,Y).
```