```
{
            frame @l2;
      @l1:
            tryf @l11;
            tryf @l101;
            copy;
            push;
            prov int(0);
            merge;
            publish;
            trye @l100002;
            cast [⟨appl(⟨term⟩)⟩,⟨appl(⟨term⟩)⟩];
            eq;
            catch @l100003;
  @l100002:
            fail;
  @l100003:
            unpublish;
            prov int(1);
            catch @l102;
    @l101:
            copy;
            push;
            prov int(1);
            merge;
            publish;
            trye @l100102;
            cast [⟨appl(⟨term⟩)⟩,⟨appl(⟨term⟩)⟩];
            eq;
            catch @l100103;
  @l100102:
            fail;
  @l100103:
            unpublish;
            prov int(1);
    @l102:
            catch @l12;
     @l11:
            copy;
            push;
            prov int(2);
            merge;
            publish;
            cast [int(⟨term⟩),int(⟨term⟩)];
            sub;
            unpublish;
            publish;
            frame @l101001;
            goto @l1;
  @l101001:
```

```
                unpublish;
                push;
                copy;
                push;
                prov int(1);
                merge;
                publish;
                cast [int(⟨term⟩),int(⟨term⟩)];
                sub;
                unpublish;
                publish;
                frame @l101011;
                goto @l1;
    @l101011:
                unpublish;
                merge;
                publish;
                cast [int(⟨term⟩),int(⟨term⟩)];
                add;
                unpublish;
        @l12:
                return;
         @l2:
                return;
}
```