Found Plan (output)

(moveto red home woods)
(tell-path red wolf woods)
(pick-up red flower woods)
(moveto wolf woods grannyhouse)
(eat-alive wolf granny grannyhouse)
(moveto red woods grannyhouse)
(eat-alive wolf red grannyhouse)
(sleep-snoreloud wolf granny red grannyhouse)
(check huntsman wolf grannyhouse)
(save huntsman wolf granny red grannyhouse)
(delivery red cakes)
(eat-together red huntsman granny cakes grannyhouse)

```
(:action moveto
  :parameters (red home woods)
  :precondition
    (and
      (forward home woods)
      (at red home)
      (not
        (stop red)
      )
    )
  :effect
    (and
      (timing red)
      (at red woods)
      (not
        (at red home)
      )
    )
)
```

Found Plan (output)

(moveto red home woods)
(tell-path red wolf woods)
(pick-up red flower woods)
(moveto wolf woods grannyhouse)
(eat-alive wolf granny grannyhouse)
(moveto red woods grannyhouse)
(eat-alive wolf red grannyhouse)
(sleep-snoreloud wolf granny red grannyhouse)
(check huntsman wolf grannyhouse)
(save huntsman wolf granny red grannyhouse)
(delivery red cakes)
(eat-together red huntsman granny cakes grannyhouse)

```
(:action tell-path
  :parameters (red wolf woods)
  :precondition
    (and
      (at red woods)
      (at wolf woods)
      (not
        (= red wolf)
      )
      (not
        (path_know red)
      )
      (path_know wolf)
    )
  :effect
    (and
      (path_know red)
      (not
        (timing red)
      )
      (stop red)
      (not
        (stop wolf)
      )
      (hungry wolf)
    )
)
```

Found Plan (output)

(moveto red home woods)

(tell-path red wolf woods)

(pick-up red flower woods)

(moveto wolf woods grannyhouse)

(eat-alive wolf granny grannyhouse)

(moveto red woods grannyhouse)

(eat-alive wolf red grannyhouse)

(sleep-snoreloud wolf granny red grannyhouse)

(check huntsman wolf grannyhouse)

(save huntsman wolf granny red grannyhouse)

(delivery red cakes)

(eat-together red huntsman granny cakes grannyhouse)

```
(:action pick-up
  :parameters (red flower woods)
  :precondition
    (and
      (at red woods)
      (errand red)
      (stop red)
      (like red flower)
    )
  :effect
    (and
      (have red flower)
      (not
        (stop red)
      )
    )
)
```

Found Plan (output)

(moveto red home woods)

(tell-path red wolf woods)

(pick-up red flower woods)

(moveto wolf woods grannyhouse)

(eat-alive wolf granny grannyhouse)

(moveto red woods grannyhouse)

(eat-alive wolf red grannyhouse)

(sleep-snoreloud wolf granny red grannyhouse)

(check huntsman wolf grannyhouse)

(save huntsman wolf granny red grannyhouse)

(delivery red cakes)

(eat-together red huntsman granny cakes grannyhouse)

```
(:action moveto
  :parameters (wolf woods grannyhouse)
  :precondition
    (and
      (forward woods grannyhouse)
      (at wolf woods)
      (not
        (stop wolf)
      )
    )
  :effect
    (and
      (timing wolf)
      (at wolf grannyhouse)
      (not
        (at wolf woods)
      )
    )
)
```

Found Plan (output)

(moveto red home woods)
(tell-path red wolf woods)
(pick-up red flower woods)
(moveto wolf woods grannyhouse)
(eat-alive wolf granny grannyhouse)
(moveto red woods grannyhouse)
(eat-alive wolf red grannyhouse)
(sleep-snoreloud wolf granny red grannyhouse)
(check huntsman wolf grannyhouse)
(save huntsman wolf granny red grannyhouse)
(delivery red cakes)
(eat-together red huntsman granny cakes grannyhouse)

```
(:action eat-alive
  :parameters (wolf granny grannyhouse)
  :precondition
    (and
      (at wolf grannyhouse)
      (at granny grannyhouse)
      (hungry wolf)
      (foodchain wolf granny)
      (timing granny)
      (alive wolf)
      (alive granny)
    )
  :effect
    (and
      (eaten granny)
      (stop granny)
    )
)
```

Found Plan (output)

(moveto red home woods)
(tell-path red wolf woods)
(pick-up red flower woods)
(moveto wolf woods grannyhouse)
(eat-alive wolf granny grannyhouse)
(moveto red woods grannyhouse)
(eat-alive wolf red grannyhouse)
(sleep-snoreloud wolf granny red grannyhouse)
(check huntsman wolf grannyhouse)
(save huntsman wolf granny red grannyhouse)
(delivery red cakes)
(eat-together red huntsman granny cakes grannyhouse)

```
(:action moveto
  :parameters (red woods grannyhouse)
  :precondition
    (and
      (forward woods grannyhouse)
      (at red woods)
      (not
        (stop red)
      )
    )
  :effect
    (and
      (timing red)
      (at red grannyhouse)
      (not
        (at red woods)
      )
    )
)
```

Found Plan (output)

(moveto red home woods)

(tell-path red wolf woods)

(pick-up red flower woods)

(moveto wolf woods grannyhouse)

(eat-alive wolf granny grannyhouse)

(moveto red woods grannyhouse)

(eat-alive wolf red grannyhouse)

(sleep-snoreloud wolf granny red grannyhouse)

(check huntsman wolf grannyhouse)

(save huntsman wolf granny red grannyhouse)

(delivery red cakes)

(eat-together red huntsman granny cakes grannyhouse)

```
(:action eat-alive
  :parameters (wolf red grannyhouse)
  :precondition
    (and
      (at wolf grannyhouse)
      (at red grannyhouse)
      (hungry wolf)
      (foodchain wolf red)
      (timing red)
      (alive wolf)
      (alive red)
    )
  :effect
    (and
      (eaten red)
      (stop red)
    )
)
```

Found Plan (output)

(moveto red home woods)

(tell-path red wolf woods)

(pick-up red flower woods)

(moveto wolf woods grannyhouse)

(eat-alive wolf granny grannyhouse)

(moveto red woods grannyhouse)

(eat-alive wolf red grannyhouse)

(sleep-snoreloud wolf granny red grannyhouse)

(check huntsman wolf grannyhouse)

(save huntsman wolf granny red grannyhouse)

(delivery red cakes)

(eat-together red huntsman granny cakes grannyhouse)

```
(:action sleep-snoreloud
  :parameters (wolf granny red grannyhouse)
  :precondition
    (and
      (at wolf grannyhouse)
      (at granny grannyhouse)
      (at red grannyhouse)
      (hungry wolf)
      (not
        (= granny red)
      )
      (eaten granny)
      (eaten red)
    )
  :effect
    (and
      (snore wolf)
      (not
        (hungry wolf)
      )
    )
)
```

Found Plan (output)

(moveto red home woods)

(tell-path red wolf woods)

(pick-up red flower woods)

(moveto wolf woods grannyhouse)

(eat-alive wolf granny grannyhouse)

(moveto red woods grannyhouse)

(eat-alive wolf red grannyhouse)

(sleep-snoreloud wolf granny red grannyhouse)

(check huntsman wolf grannyhouse)

(save huntsman wolf granny red grannyhouse)

(delivery red cakes)

(eat-together red huntsman granny cakes grannyhouse)

```
(:action check
  :parameters (huntsman wolf grannyhouse)
  :precondition
    (and
      (at huntsman grannyhouse)
      (at wolf grannyhouse)
      (not
        (= huntsman wolf)
      )
      (not
        (eaten huntsman)
      )
      (snore wolf)
    )
  :effect
    (and
      (checked huntsman)
    )
)
```

Found Plan (output)

(moveto red home woods)

(tell-path red wolf woods)

(pick-up red flower woods)

(moveto wolf woods grannyhouse)

(eat-alive wolf granny grannyhouse)

(moveto red woods grannyhouse)

(eat-alive wolf red grannyhouse)

(sleep-snoreloud wolf granny red grannyhouse)

(check huntsman wolf grannyhouse)

(save huntsman wolf granny red grannyhouse)

(delivery red cakes)

(eat-together red huntsman granny cakes grannyhouse)

```
(:action save
  :parameters (huntsman wolf granny red grannyhouse)
  :precondition
    (and
      (at huntsman grannyhouse)
      (at wolf grannyhouse)
      (at granny grannyhouse)
      (at red grannyhouse)
      (checked huntsman)
      (snore wolf)
      (eaten granny)
      (eaten red)
      (not
        (= granny red)
      )
    )
  :effect
    (and
      (not
        (alive wolf)
      )
      (saved granny)
      (saved red)
    )
)
```

Found Plan (output)

(moveto red home woods)
(tell-path red wolf woods)
(pick-up red flower woods)
(moveto wolf woods grannyhouse)
(eat-alive wolf granny grannyhouse)
(moveto red woods grannyhouse)
(eat-alive wolf red grannyhouse)
(sleep-snoreloud wolf granny red grannyhouse)
(check huntsman wolf grannyhouse)
(save huntsman wolf granny red grannyhouse)
(delivery red cakes)
(eat-together red huntsman granny cakes grannyhouse)

```
(:action delivery
  :parameters (red cakes)
  :precondition
    (and
      (saved red)
      (errand red)
    )
  :effect
    (and
      (have red cakes)
    )
)
```

Found Plan (output)

(moveto red home woods)
(tell-path red wolf woods)
(pick-up red flower woods)
(moveto wolf woods grannyhouse)
(eat-alive wolf granny grannyhouse)
(moveto red woods grannyhouse)
(eat-alive wolf red grannyhouse)
(sleep-snoreloud wolf granny red grannyhouse)
(check huntsman wolf grannyhouse)
(save huntsman wolf granny red grannyhouse)
(delivery red cakes)
(eat-together red huntsman granny cakes grannyhouse)

```
(:action eat-together
  :parameters (red huntsman granny cakes grannyhouse)
  :precondition
    (and
      (at red grannyhouse)
      (at huntsman grannyhouse)
      (at granny grannyhouse)
      (have red cakes)
      (not
        (happy red)
      )
      (not
        (= huntsman granny)
      )
      (like huntsman cakes)
      (like granny cakes)
    )
  :effect
    (and
      (happy red)
      (happy huntsman)
      (happy granny)
    )
)
```