

# Exercise

```
deferred class
  ITERABLE [G]
  feature -- Access
    new_cursor: ITERATION_CURSOR [G]
  deferred end
end
```

new\_cursor\*

```
deferred class
  ITERATION_CURSOR [G]
  feature -- Cursor features
    item: G
  deferred end

  after: BOOLEAN
  deferred end

  forth
  deferred end
```

```
test_database: BOOLEAN
local
  db: DATABASE[STRING, INTEGER]
  tuples: LINKED_LIST[TUPLE[INTEGER, STRING]]
do
  create db.make
  create tuples.make
across
  db is t
loop
  tuples.extend (t)
end
end
```

Solution

```
class
  DATABASE[G, H]
inherit
  ITERABLE[TUPLE[H, G]]
feature {NONE} -- Implementation
  gs: ARRAY[G]
  hs: ARRAY[H]
feature -- Iterable
  new_cursor: ITERATION_CURSOR[TUPLE[H, G]]
  local
    db_cursor: ITEM_ITERATION_CURSOR[H, G]
  do
    create db_cursor.make (gs, hs)
  Result := db_cursor
  end
end
```

new\_cursor+

```
class
  ITEM_ITERATION_CURSOR[M, N]
inherit
  ITERATION_CURSOR[TUPLE[M, N]]
create
  make
feature {NONE} -- Implementation
  ms: ARRAY[M]
  ns: ARRAY[N]
feature -- Constructor
  make (new_ns: ARRAY[N]; new_ms: ARRAY[M])
  do ... end
feature -- Cursor features
  item: TUPLE[M, N]
  do ... end

  after: BOOLEAN
  do ... end

  forth
  do ... end
end
```

+db+