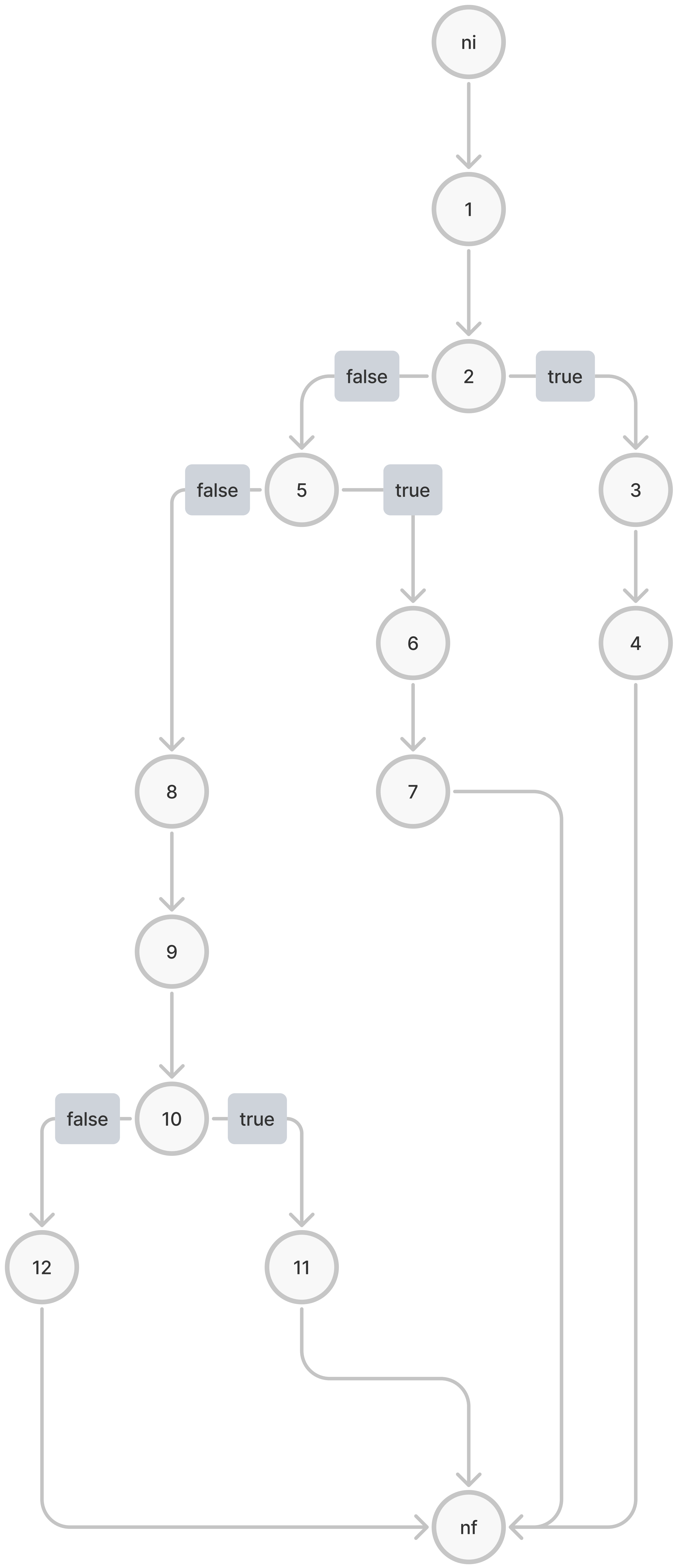

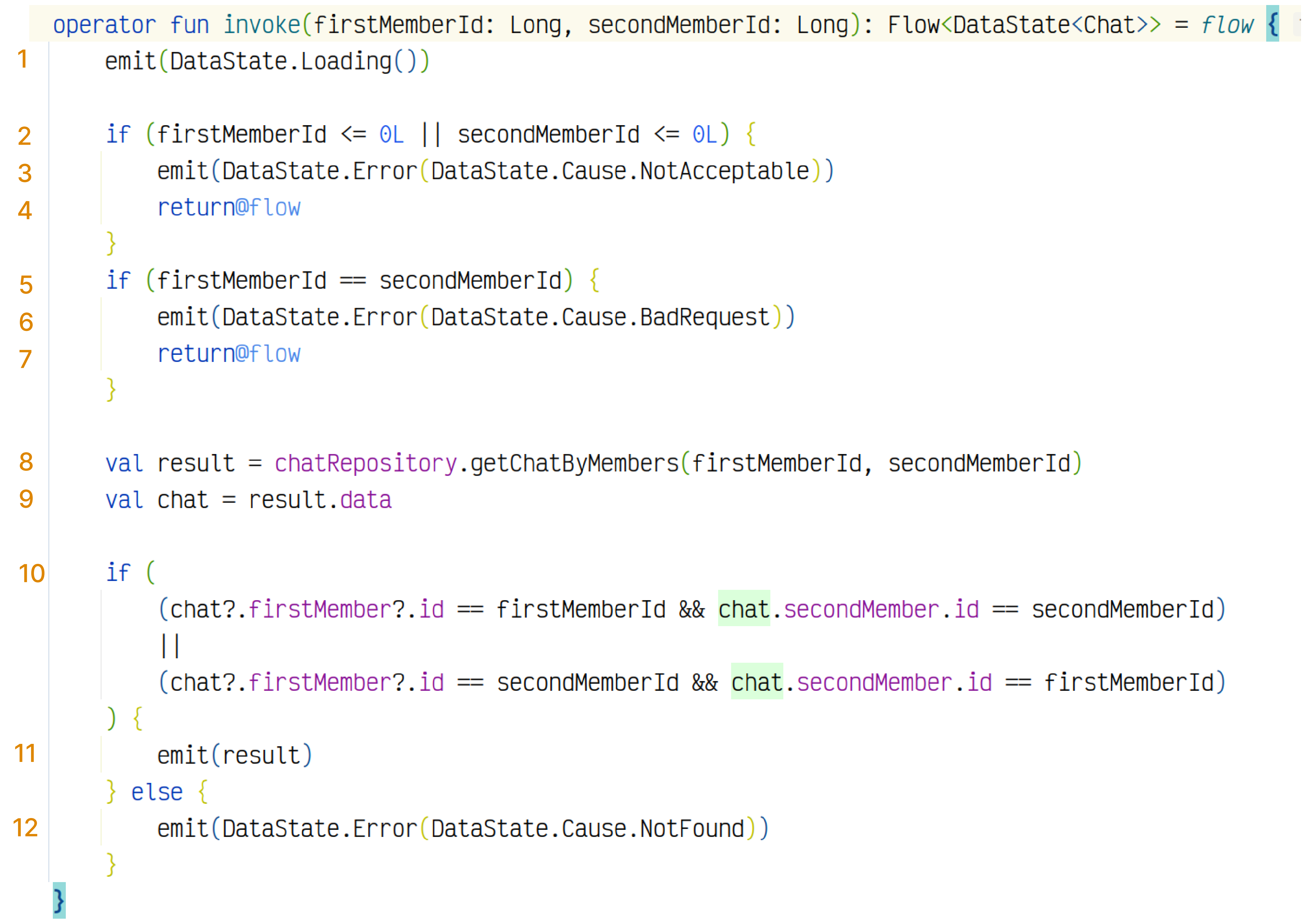

```
    operator fun invoke(firstMemberId: Long, secondMemberId: Long): Flow<DataState<Chat>> = flow {
1       emit(DataState.Loading())

2       if (firstMemberId <= 0L || secondMemberId <= 0L) {
3           emit(DataState.Error(DataState.Cause.NotAcceptable))
4           return@flow
        }
5       if (firstMemberId == secondMemberId) {
6           emit(DataState.Error(DataState.Cause.BadRequest))
7           return@flow
        }

8       val result = chatRepository.getChatByMembers(firstMemberId, secondMemberId)
9       val chat = result.data

10      if (
            (chat?.firstMember?.id == firstMemberId && chat.secondMember.id == secondMemberId)
            ||
            (chat?.firstMember?.id == secondMemberId && chat.secondMember.id == firstMemberId)
        ) {
11          emit(result)
        } else {
12          emit(DataState.Error(DataState.Cause.NotFound))
        }
    }
```