# 6. Writing contracts for noteworthy classes

CampaignMgr class requires the attribute maxNumCampaigns to be positive.

context CampaignMgr inv.

self.getMaxNumCampains() > 0

acceptCampagin(c) assumes that c has not yet been accepted in the CampaignMgr

context CampaignMgr::acceptCampaign(c) pre:

!isCampaignNegotiated(c)

State that the CampaignMgr must not yet have reached the maximum number of Campaigns before invoking acceptCampaign().

context CampaignMgr:acceptCampaign(c) pre:

getNumCampaigns() < getMaxNumCampaigns()

State that the Campaign c should be known to the CampaginMgr after acceptCampaign() returns.

context CampaignMgr::acceptCampaign(c) post:

isCampaignNegotiated(c)

State that number of Campaigns in the CampaignMgr increases by one with the invocation of acceptCampaign().

context CampaignMgr::acceptCampaign(c) post:

getNumCampaigns() = @pre.getNumCampaigns() + 1

Invoke getActiveCampaigns() from AdvertConsult results in all ongoing Campaigns, where each Campaign contains a Director and a Client, and the Staffs who are assigned with.

context AdvertConsult::getActiveCampaigns post:

result = campaigns -> asSet

Ensure all Advertisements in a Campaign occur within the Campaign’s time frame.

context Campagin inv:

advertisements -> forAll(a:Advertisement | a.start.after(c.start) and a.end.before(c.end)