1*2*129[false]: cp:int=0, cbalance:int=D, gp:int=0, gbalance:int=D, cur_q:int=q:int, ID:human_address=id:human_address
C(garbage_bin):ID:human_address.OK::int(R)
(cur_q:int = 1)
1*3*108[true]: cp:int=0, cbalance:int=D, gp:int=1, gbalance:int=(D + -R)
C(garbage_bin):ID:human_address.NOK::()
(cur_q:int > 2)
C(garbage_bin).dep::int*human_address[](q':int,id':human_address)
1*4*110[false]: cp:int=0, cbalance:int=D, gp:int=1, gbalance:int=(D + -R), cur_q:int=q':int, ID:human_address=id':human_address
C(garbage_bin):ID:human_address.OK::int(R)
(cur_q:int = 1)
C(garbage_bin):ID:human_address.OK::int((2 * R))
(cur_q:int = 2)
1*5*71[true]: cp:int=0, cbalance:int=D, gp:int=2, gbalance:int=(D + -(2 * R))
C(garbage_bin):ID:human_address.lost::int(of:int)
((2 * R:int) > of:int)
C(garbage_bin).bid::int*human_address[](e:int,gt_id:human_address)
1*6*73[true]: cp:int=0, cbalance:int=D, gp:int=2, gbalance:int=(D + -(2 * R)), of:int=e':int, ID:human_address=gt_id:human_address
C(garbage_bin).bid::int*human_address[](e':int,gt_id':human_address)
(of:int >= (2 * R:int))
1*7*76[true]: cp:int=0, cbalance:int=D, gp:int=2, gbalance:int=(D + -(2 * R)), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
C(garbage_bin):ID':human_address.LOST::int(of':int)
(of:int >= of':int)
C(garbage_bin):ID:human_address.LOST::int(of:int)
(of:int > of':int)
1*8*78[true]: cp:int=0, cbalance:int=D, gp:int=2, gbalance:int=(D + -(2 * R)), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
C(garbage_bin):ID:human_address.WIN::()
(of:int >= of':int)
C(garbage_bin):ID':human_address.WIN::()
(of:int >= of':int)
1*9*80[true]: cp:int=0, cbalance:int=D, gp:int=2, gbalance:int=(D + -(2 * R)), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
tau
(((of:int >= of':int) /\ ~(id:human_address = ID:human_address)) \/ ((of':int > of:int) /\ ~(id:human_address = ID':human_address)))
C(garbage_bin).empty::string[](id:string)
1*10*82[false]: cp:int=0, cbalance:int=D, gp:int=0, gbalance:int=(D + -(2 * R)), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
C(garbage_bin):incinerator:contract_address.notify::string*int(ID:string,id:int)
(of':int >= of:int)
C(garbage_bin):incinerator:contract_address.notify::string*int(ID:string,id:int)
(of:int >= of':int)
1*11*84[true]: cp:int=0, cbalance:int=D, gp:int=0, gbalance:int=(D + (-(2 * R) + (max of:int of':int))), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
H(basiccitizen):C(incinerator).fee::int(D:int)
2*11*47[true]: cp:int=2, cbalance:int=0, gp:int=0, gbalance:int=(D + (-(2 * R) + (max of:int of':int))), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
C(garbage_bin):incinerator:contract_address.save::int((D:int + (-(2 * R) + (max of:int of':int))))
2*1*33[true]: cp:int=2, cbalance:int=0, gp:int=0, gbalance:int=D
tau
C(garbage_bin):ID:human_address.NOK::()
(cur_q:int > 2)
C(garbage_bin).dep::int*human_address[](q:int,id:human_address)
3*2*127[false]: cp:int=1, cbalance:int=0, gp:int=0, gbalance:int=D, cur_q:int=1, ID:human_address=H(basiccitizen)
2*2*34[false]: cp:int=2, cbalance:int=0, gp:int=0, gbalance:int=D, cur_q:int=q:int, ID:human_address=id:human_address
tau
C(garbage_bin):ID:human_address.OK::int(R)
(cur_q:int = 1)
H(basiccitizen):C(incinerator).fee::int(D:int)
4*3*65[true]: cp:int=1, cbalance:int=R, gp:int=1, gbalance:int=(D + -R)
2*3*36[true]: cp:int=2, cbalance:int=0, gp:int=1, gbalance:int=(D + -R)
tau
C(garbage_bin):ID:human_address.NOK::()
(cur_q:int > 2)
C(garbage_bin).dep::int*human_address[](q':int,id':human_address)
C(garbage_bin):ID:human_address.OK::int((2 * R))
(cur_q:int = 2)
tau
C(garbage_bin).dep::int*human_address[](q':int,id':human_address)
C(garbage_bin):ID:human_address.NOK::()
(cur_q:int > 2)
5*4*69[false]: cp:int=0, cbalance:int=R, gp:int=1, gbalance:int=(D + -R), cur_q:int=1, ID:human_address=H(basiccitizen)
4*4*66[false]: cp:int=1, cbalance:int=R, gp:int=1, gbalance:int=(D + -R), cur_q:int=q':int, ID:human_address=id':human_address
3*4*52[false]: cp:int=1, cbalance:int=0, gp:int=1, gbalance:int=(D + -R), cur_q:int=1, ID:human_address=H(basiccitizen)
2*4*37[false]: cp:int=2, cbalance:int=0, gp:int=1, gbalance:int=(D + -R), cur_q:int=q':int, ID:human_address=id':human_address
tau
C(garbage_bin):ID:human_address.OK::int(R)
(cur_q:int = 1)
tau
C(garbage_bin):ID:human_address.OK::int(R)
(cur_q:int = 1)
C(garbage_bin):incinerator:contract_address.notify::string*int(ID:string,id:int)
(of':int >= of:int)
C(garbage_bin):incinerator:contract_address.notify::string*int(ID:string,id:int)
(of:int >= of':int)
H(basiccitizen):C(banca).save::int(balance:int)
6*5*70[true]: cp:int=0, cbalance:int=R, gp:int=2, gbalance:int=(D + -(2 * R))
4*5*53[true]: cp:int=1, cbalance:int=R, gp:int=2, gbalance:int=(D + -(2 * R))
2*5*39[true]: cp:int=2, cbalance:int=0, gp:int=2, gbalance:int=(D + -(2 * R))
C(garbage_bin).bid::int*human_address[](e:int,gt_id:human_address)
C(garbage_bin):ID:human_address.lost::int(of:int)
((2 * R:int) > of:int)
C(garbage_bin):ID:human_address.lost::int(of:int)
((2 * R:int) > of:int)
C(garbage_bin).bid::int*human_address[](e:int,gt_id:human_address)
C(garbage_bin).bid::int*human_address[](e:int,gt_id:human_address)
C(garbage_bin):ID:human_address.lost::int(of:int)
((2 * R:int) > of:int)
6*6*90[true]: cp:int=0, cbalance:int=R, gp:int=2, gbalance:int=(D + -(2 * R)), of:int=e':int, ID:human_address=gt_id:human_address
4*6*54[true]: cp:int=1, cbalance:int=R, gp:int=2, gbalance:int=(D + -(2 * R)), of:int=e':int, ID:human_address=gt_id:human_address
2*6*40[true]: cp:int=2, cbalance:int=0, gp:int=2, gbalance:int=(D + -(2 * R)), of:int=e':int, ID:human_address=gt_id:human_address
C(garbage_bin).bid::int*human_address[](e':int,gt_id':human_address)
(of:int >= (2 * R:int))
C(garbage_bin).bid::int*human_address[](e':int,gt_id':human_address)
(of:int >= (2 * R:int))
C(garbage_bin).bid::int*human_address[](e':int,gt_id':human_address)
(of:int >= (2 * R:int))
6*7*93[true]: cp:int=0, cbalance:int=R, gp:int=2, gbalance:int=(D + -(2 * R)), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
4*7*56[true]: cp:int=1, cbalance:int=R, gp:int=2, gbalance:int=(D + -(2 * R)), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
2*7*42[true]: cp:int=2, cbalance:int=0, gp:int=2, gbalance:int=(D + -(2 * R)), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
C(garbage_bin):ID':human_address.LOST::int(of':int)
(of:int >= of':int)
C(garbage_bin):ID:human_address.LOST::int(of:int)
(of:int > of':int)
C(garbage_bin):ID:human_address.LOST::int(of:int)
(of:int > of':int)
C(garbage_bin):ID':human_address.LOST::int(of':int)
(of:int >= of':int)
C(garbage_bin):ID':human_address.LOST::int(of':int)
(of:int >= of':int)
C(garbage_bin):ID:human_address.LOST::int(of:int)
(of:int > of':int)
6*8*95[true]: cp:int=0, cbalance:int=R, gp:int=2, gbalance:int=(D + -(2 * R)), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
4*8*57[true]: cp:int=1, cbalance:int=R, gp:int=2, gbalance:int=(D + -(2 * R)), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
2*8*43[true]: cp:int=2, cbalance:int=0, gp:int=2, gbalance:int=(D + -(2 * R)), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
C(garbage_bin):ID:human_address.WIN::()
(of:int >= of':int)
C(garbage_bin):ID':human_address.WIN::()
(of:int >= of':int)
C(garbage_bin):ID':human_address.WIN::()
(of:int >= of':int)
C(garbage_bin):ID:human_address.WIN::()
(of:int >= of':int)
C(garbage_bin):ID:human_address.WIN::()
(of:int >= of':int)
C(garbage_bin):ID':human_address.WIN::()
(of:int >= of':int)
6*9*97[true]: cp:int=0, cbalance:int=R, gp:int=2, gbalance:int=(D + -(2 * R)), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
4*9*58[true]: cp:int=1, cbalance:int=R, gp:int=2, gbalance:int=(D + -(2 * R)), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
2*9*44[true]: cp:int=2, cbalance:int=0, gp:int=2, gbalance:int=(D + -(2 * R)), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
C(garbage_bin):ID:human_address.OK::int((2 * R))
(cur_q:int = 2)
C(garbage_bin):ID:human_address.OK::int((2 * R))
(cur_q:int = 2)
C(garbage_bin):ID:human_address.OK::int(R)
(cur_q:int = 1)
C(garbage_bin):ID:human_address.OK::int(R)
(cur_q:int = 1)
tau
(((of:int >= of':int) /\ ~(id:human_address = ID:human_address)) \/ ((of':int > of:int) /\ ~(id:human_address = ID':human_address)))
C(garbage_bin).empty::string[](id:string)
tau
(((of:int >= of':int) /\ ~(id:human_address = ID:human_address)) \/ ((of':int > of:int) /\ ~(id:human_address = ID':human_address)))
C(garbage_bin).empty::string[](id:string)
tau
(((of:int >= of':int) /\ ~(id:human_address = ID:human_address)) \/ ((of':int > of:int) /\ ~(id:human_address = ID':human_address)))
C(garbage_bin).empty::string[](id:string)
6*10*99[false]: cp:int=0, cbalance:int=R, gp:int=0, gbalance:int=(D + -(2 * R)), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
4*10*59[false]: cp:int=1, cbalance:int=R, gp:int=0, gbalance:int=(D + -(2 * R)), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
2*10*45[false]: cp:int=2, cbalance:int=0, gp:int=0, gbalance:int=(D + -(2 * R)), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
C(garbage_bin):incinerator:contract_address.notify::string*int(ID:string,id:int)
(of:int >= of':int)
C(garbage_bin):incinerator:contract_address.notify::string*int(ID:string,id:int)
(of':int >= of:int)
C(garbage_bin):incinerator:contract_address.notify::string*int(ID:string,id:int)
(of:int >= of':int)
C(garbage_bin):incinerator:contract_address.notify::string*int(ID:string,id:int)
(of':int >= of:int)
6*11*101[true]: cp:int=0, cbalance:int=R, gp:int=0, gbalance:int=(D + (-(2 * R) + (max of:int of':int))), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
4*11*61[true]: cp:int=1, cbalance:int=R, gp:int=0, gbalance:int=(D + (-(2 * R) + (max of:int of':int))), of:int=e:int, ID:human_address=gt_id:human_address, of':int=e':int, ID':human_address=gt_id':human_address
C(garbage_bin):incinerator:contract_address.save::int((D:int + (-(2 * R) + (max of:int of':int))))
C(garbage_bin):incinerator:contract_address.save::int((D:int + (-(2 * R) + (max of:int of':int))))
6*1*103[true]: cp:int=0, cbalance:int=R, gp:int=0, gbalance:int=D
4*1*62[true]: cp:int=1, cbalance:int=R, gp:int=0, gbalance:int=D
C(garbage_bin):ID:human_address.NOK::()
(cur_q:int > 2)
C(garbage_bin).dep::int*human_address[](q:int,id:human_address)
C(garbage_bin).dep::int*human_address[](q:int,id:human_address)
C(garbage_bin):ID:human_address.NOK::()
(cur_q:int > 2)
tau
H(basiccitizen):C(banca).save::int(balance:int)
6*2*105[false]: cp:int=0, cbalance:int=R, gp:int=0, gbalance:int=D, cur_q:int=q:int, ID:human_address=id:human_address
4*2*63[false]: cp:int=1, cbalance:int=R, gp:int=0, gbalance:int=D, cur_q:int=q:int, ID:human_address=id:human_address
5*2*121[false]: cp:int=0, cbalance:int=R, gp:int=0, gbalance:int=D, cur_q:int=1, ID:human_address=H(basiccitizen)
1*1*32[true]: cp:int=0, cbalance:int=D, gp:int=0, gbalance:int=D
C(garbage_bin):ID:human_address.OK::int(R)
(cur_q:int = 1)
tau
6*3*107[true]: cp:int=0, cbalance:int=R, gp:int=1, gbalance:int=(D + -R)
C(garbage_bin):ID:human_address.NOK::()
(cur_q:int > 2)
C(garbage_bin).dep::int*human_address[](q':int,id':human_address)
6*4*113[false]: cp:int=0, cbalance:int=R, gp:int=1, gbalance:int=(D + -R), cur_q:int=q':int, ID:human_address=id':human_address
H(basiccitizen):C(banca).save::int(balance:int)
H(basiccitizen):C(incinerator).fee::int(D:int)
C(garbage_bin).dep::int*human_address[](q:int,id:human_address)
C(garbage_bin):ID:human_address.NOK::()
(cur_q:int > 2)