

به نام خدا

تمرین سری سوم
مهندسی نرم افزار ۲

نعیم اصفهانی
۸۴۲۰۱۰۰۳

1.

function the_same(Stack S1, Stack S2, int s) : Boolean b;

pre: $s > 0$

post: $((\forall i \in \mathbb{N}, i \leq s : S1[i] = S2[i]) \wedge b = \text{true}) \otimes$
 $((\exists i \in \mathbb{N}, i \leq s : S1[i] \neq S2[i]) \wedge b = \text{false})$
 $\wedge S1' = S1 \wedge S2' = S2$

err: $b = \text{true} \wedge S1' = S1 \wedge S2' = S2$

function init() : Stack S;

pre: true

post: isEmpty(S) = true

function push(Stack S, CARDINAL n) : Stack S2;

pre: true

post: $S2.\text{length} = S.\text{length} + 1 \wedge S' = S \wedge \text{the_same}(S, S2, S.\text{length})$
 $\wedge \text{top}(S2) = n$

function pop(Stack S) : Stack S2;

pre: ! isEmpty(S)

post: $S2.\text{length} = S.\text{length} - 1 \wedge S' = S \wedge \text{the_same}(S, S2, S2.\text{length})$

err: $S' = S \wedge S2 = S$

function top(Stack S) : CARDINAL n;

pre: ! isEmpty(S)

post: $S[S.\text{length}] = n \wedge S' = S$

err: $S' = S \wedge n = 0$

function isEmpty(Stack S) : Boolean b;

pre: true

post: $((S.\text{length} = 0 \wedge b = \text{true}) \otimes (S.\text{length} \neq 0 \wedge b = \text{false}))$
 $\wedge S.\text{length} \geq 0 \wedge S' = S$

2.

STACK(CARDINAL : [Undefined → CARDINAL])

sort Stack
importsCARDINAL, BOOLEAN

init → Stack;
push(Stack, Cardinal) → Stack;
pop(Stack) → Stack;
top(Stack) → Cardinal;
isEmpty(Stack) → Boolean;

top(init) = Undefined
isEmpty(init) = true
pop(init) = Undefined
top(push(s, n)) = n
isEmpty(push(s, n)) = false
pop(push(s, n)) = s

3.

$[Person, Car]$

$Type ::= Small \mid Medium \mid Big$

$carrep : Car \leftrightarrow Type$
 $SCars : seq\ Car$
 $MCars : seq\ Car$
 $BCars : seq\ Car$

$CarSys$
 $users : \{p \mid p \in Person\}$
 $rental : Car \leftrightarrow Person$

$CarSysInit$
 $CarSys'$
 $users' = \emptyset$
 $rental' = \emptyset$

$RentSCar$
 $\Delta CarSys$
 $u? : Person$
 $t? : Type$
 $c! : Car$
 $t? = Small$
 $(\#SCars \neq 0 \wedge c! = head\ SCars \wedge SCars' = tail\ SCars \wedge rental' = rental \cup (c! \mapsto u?)) \otimes (c! = Undefined \wedge \#SCars = 0 \wedge SCars' = SCars)$

$ReturnSCar$
 $\Delta CarSys$
 $c? : Car$
 $(c?, Small) \in carrep$
 $c? \in dom\ rental$
 $rental' = \{c?\} \triangleleft rental$
 $last\ SCars' = c?$
 $front\ SCars' = SCars$

RentMCar

$\Delta CarSys$

$u? : Person$

$t? : Type$

$c! : Car$

$t? = Medium$

$(\#MCars \neq 0 \wedge c! = head\ MCars \wedge MCars' = tail\ MCars \wedge rental' = rental \cup (c! \mapsto u?)) \otimes (c! = Undefined \wedge \#MCars = 0 \wedge MCars' = MCars)$

ReturnMCar

$\Delta CarSys$

$c? : Car$

$(c?, Medium) \in carrep$

$c? \in dom\ rental$

$rental' = \{c?\} \triangleleft rental$

$last\ MCars' = c?$

$front\ MCars' = MCars$

RentBCar

$\Delta CarSys$

$u? : Person$

$t? : Type$

$c! : Car$

$t? = Big$

$(\#BCars \neq 0 \wedge c! = head\ BCars \wedge BCars' = tail\ BCars \wedge rental' = rental \cup (c! \mapsto u?)) \otimes (c! = Undefined \wedge \#BCars = 0 \wedge BCars' = BCars)$

ReturnBCar

$\Delta CarSys$

$c? : Car$

$(c?, Big) \in carrep$

$c? \in dom\ rental$

$rental' = \{c?\} \triangleleft rental$

$last\ BCars' = c?$

$front\ BCars' = BCars$

AddUser ΔCarSys $u? : \text{Person}$
$users' = users \cup \{u?\}$

ListUsers ΞCarSys $t? : \text{Type}$ $! : \mathbb{P} \text{Person}$
$! = \{p \mid \exists c : (c, t?) \in \text{carrep} \wedge (c, p) \in \text{rental}\}$

$\text{RentCar} \cong \text{AddUser} \wedge (\text{RentSCar} \vee \text{RentMCar} \vee \text{RentBCar})$
 $\text{ReturnCar} \cong \text{ReturnSCar} \vee \text{ReturnMCar} \vee \text{ReturnBCar}$

4.

در این عمل در فایل داده‌ی مربوط به یک کلید موجود به روز می‌شود. با توجه به این‌که رابطه‌ی محتوا بودن یک ترتیب جزئی است، ممکن است همه‌ی کلیدها در دامنه‌ی آن نباشند؛ در این عمل به یکی از کلیدهایی که در دامنه‌ی این ترتیب است یک مقدار جدید می‌دهیم. داده متناظر با کلید مربوطه که در دامنه‌ی محتوا حضور دارد در فایل به روز می‌شود و پیغام “okay” داده می‌شود. در صورت وجود نداشتن کلید مربوط به اطلاع داده شده در دامنه‌ی محتوا عمل نوشتن انجام نمی‌شود و پیغام “key_not_in_use” داده می‌شود.