Bhavesh Rajesh Talreja (A20516822)
CS 536 HW6

1. Loop Bounds and Full Proof Outlines

Task 1.1

Given Program:

```
                                   $[gas_0 = gas \land gas > 0 \land batt = 0]$
$miles := \bar{0}$;
{inv ____ $\land\, gas \geq 0 \land batt \geq 0$}
{dec ____}
while $gas > 1 \lor batt > 0$ do
   if $batt > 0$ then
      $batt := batt - 1$
   else
      $gas := gas - 2$;
      $batt := batt + 1$
   fi;
   $miles := miles + 1$
od
                                   $[miles \geq gas_0 - 1]$
```

Ans.

We are asked to find an invariant, a decreasing clause/ a loop bound.

Also, we need to convert this into a full proof outline. Prove the termination along with partial correctness.

The invariant that worked for me is

{inv P $\triangleq$ $miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0$}

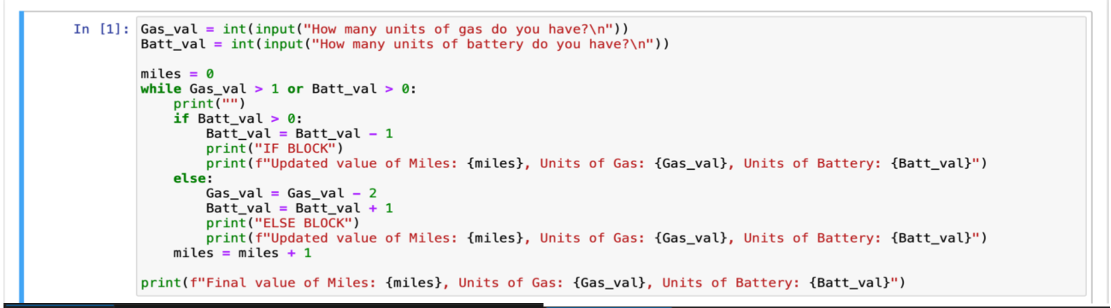
Initially I found {inv P $\triangleq$ $miles + gas + batt \geq gas_0 \land gas \geq 0 \land batt \geq 0$} which is same as the above one on Dafny. But found the above one to be easy to prove while writing the proof below.

The decreasing clause that worked for me is

{dec $gas + batt$}

Bhavesh Rajesh Talreja (A20516822)
CS 536 HW6

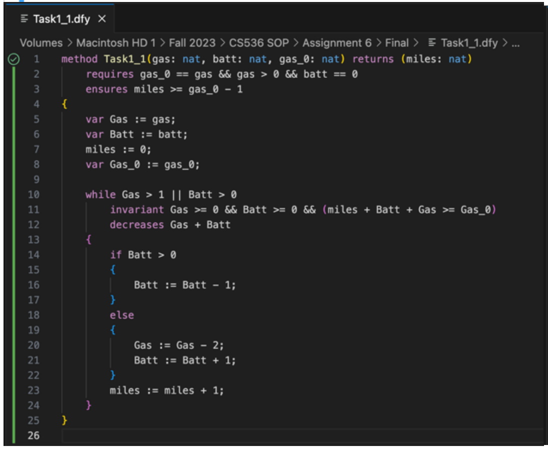
The idea for the invariant was inspired when I tried to execute this program in python.



```
In [1]: Gas_val = int(input("How many units of gas do you have?\n"))
        Batt_val = int(input("How many units of battery do you have?\n"))

        miles = 0
        while Gas_val > 1 or Batt_val > 0:
            print("")
            if Batt_val > 0:
                Batt_val = Batt_val - 1
                print("IF BLOCK")
                print(f"Updated value of Miles: {miles}, Units of Gas: {Gas_val}, Units of Battery: {Batt_val}")
            else:
                Gas_val = Gas_val - 2
                Batt_val = Batt_val + 1
                print("ELSE BLOCK")
                print(f"Updated value of Miles: {miles}, Units of Gas: {Gas_val}, Units of Battery: {Batt_val}")
            miles = miles + 1

        print(f"Final value of Miles: {miles}, Units of Gas: {Gas_val}, Units of Battery: {Batt_val}")
```

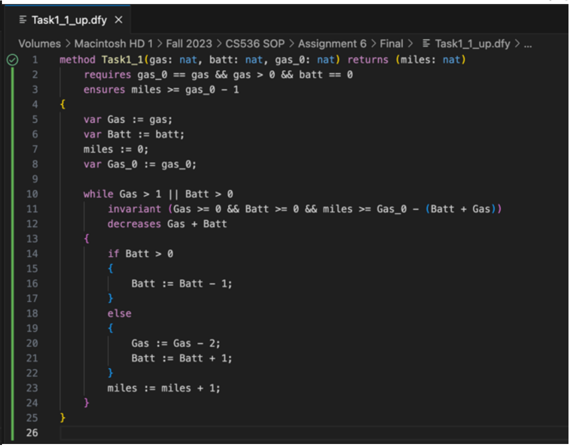


Task1_1.dfy ×
Volumes > Macintosh HD 1 > Fall 2023 > CS536 SOP > Assignment 6 > Final > Task1_1.dfy > ...

```
method Task1_1(gas: nat, batt: nat, gas_0: nat) returns (miles: nat)
    requires gas_0 == gas && gas > 0 && batt == 0
    ensures miles >= gas_0 - 1
{
    var Gas := gas;
    var Batt := batt;
    miles := 0;
    var Gas_0 := gas_0;

    while Gas > 1 || Batt > 0
        invariant Gas >= 0 && Batt >= 0 && (miles + Batt + Gas >= Gas_0)
        decreases Gas + Batt
    {
        if Batt > 0
        {
            Batt := Batt - 1;
        }
        else
        {
            Gas := Gas - 2;
            Batt := Batt + 1;
        }
        miles := miles + 1;
    }
}

```



Task1_1_up.dfy ×
Volumes > Macintosh HD 1 > Fall 2023 > CS536 SOP > Assignment 6 > Final > Task1_1_up.dfy > ...

```
method Task1_1(gas: nat, batt: nat, gas_0: nat) returns (miles: nat)
    requires gas_0 == gas && gas > 0 && batt == 0
    ensures miles >= gas_0 - 1
{
    var Gas := gas;
    var Batt := batt;
    miles := 0;
    var Gas_0 := gas_0;

    while Gas > 1 || Batt > 0
        invariant (Gas >= 0 && Batt >= 0 && miles >= Gas_0 - (Batt + Gas))
        decreases Gas + Batt
    {
        if Batt > 0
        {
            Batt := Batt - 1;
        }
        else
        {
            Gas := Gas - 2;
            Batt := Batt + 1;
        }
        miles := miles + 1;
    }
}

```

Now, we will convert this minimal proof outline into a full proof outline.

Partial correctness:

```
                                    $\{gas_0 = gas \ \wedge gas > 0 \ \wedge batt = 0\}$
$miles := \bar{0};$
{inv P ≜ $miles \geq \ gas_0 - (gas + batt) \wedge gas \geq 0 \ \wedge batt \geq 0\}$
{dec $(gas + batt)$}
while $gas > 1 \ \vee batt > 0$ do
  if $batt > 0$ then
    $batt := batt - 1$
  else
    $gas := gas - 2;$
    $batt := batt + 1$
  fi;
  $miles := miles + 1$
od
                                    $\{miles \geq \ gas_0 - 1\}$
```

Applying rule 9,
Add conditions to a loop based on the invariant:

{inv P} while e do S od
becomes
{inv P} while e do {P ∧ e} S {P} od {P ∧ ¬e}

```
                              $\{gas_0 = gas \ \wedge gas > 0 \ \wedge batt = 0\}$
$miles := \bar{0};$
{inv P $\triangleq$ $miles \geq \ gas_0 - (gas + batt) \wedge gas \geq 0 \ \wedge batt \geq 0\}$
{dec $(gas + batt)\}$
while $gas > 1 \ \vee batt > 0$ do $\{miles \geq \ gas_0 - (gas + batt) \wedge gas \geq 0 \ \wedge batt \geq 0 \wedge$
                                                  $gas > 1 \ \vee batt > 0\}$
  if $batt > 0$ then
    $batt := batt - 1$
  else
    $gas := gas - 2;$
    $batt := batt + 1$
  fi;
  $miles := miles + 1$        $\{miles \geq \ gas_0 - (gas + batt) \wedge gas \geq 0 \ \wedge batt \geq 0\}$
od    $\{miles \geq \ gas_0 - (gas + batt) \wedge gas \geq 0 \ \wedge batt \geq 0 \wedge gas \leq 1 \ \wedge batt \leq 0\}$
                              $\{miles \geq \ gas_0 - 1\}$
```

We have a proof obligation here:

$\{miles \geq \ gas_0 - (gas + batt) \wedge gas \geq 0 \ \wedge batt \geq 0 \wedge gas \leq 1 \ \wedge batt \leq 0\}$
$\{miles \geq \ gas_0 - 1\}$

$\{miles \geq \ gas_0 - (gas + batt) \wedge gas \geq 0 \ \wedge batt \geq 0 \wedge gas \leq 1 \ \wedge batt \leq 0\}$
$\Rightarrow \{miles \geq \ gas_0 - (gas + batt) \wedge gas \geq 0 \wedge gas \leq 1 \wedge batt = 0\}$
$\Rightarrow \{miles \geq \ gas_0 - (gas + batt) \wedge (gas = 0 \vee gas = 1) \wedge batt = 0\}$

When gas=0 and batt=0
$\Rightarrow \{miles \geq \ gas_0\} \Rightarrow \{miles \geq \ gas_0 - 1\}$

When gas=1 and batt=0
$\Rightarrow \{miles \geq \ gas_0 - 1\}$

So, we can rewrite our proof outline as:

```
                              $\{gas_0 = gas \ \wedge gas > 0 \ \wedge batt = 0\}$
$miles := \bar{0};$
{inv P $\triangleq$ $miles \geq \ gas_0 - (gas + batt) \wedge gas \geq 0 \ \wedge batt \geq 0\}$
{dec $(gas + batt)\}$
while $gas > 1 \ \vee batt > 0$ do $\{miles \geq \ gas_0 - (gas + batt) \wedge gas \geq 0 \ \wedge batt \geq 0 \wedge$
                                                  $gas > 1 \ \vee batt > 0\}$
```

```
  if $batt > 0$ then
    $batt := batt - 1$
  else
    $gas := gas - 2;$
    $batt := batt + 1$
  fi;
  $miles := miles + 1$          $\{miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$
od     $\{miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge gas \leq 1 \wedge batt \leq 0\}$
                                        $\Rightarrow\{miles \geq gas_0 - 1\}$
```

Here, we will use rule 1:

Prepend {wlp(x := e, Q)} to x := e {Q}.

```
                                        $\{gas_0 = gas \wedge gas > 0 \wedge batt = 0\}$
$miles := \overline{0};$
{inv P $\triangleq$ $miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$
{dec $(gas + batt)\}$
while $gas > 1 \vee batt > 0$ do $\{miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge$
                                                    $gas > 1 \vee batt > 0\}$
  if $batt > 0$ then
    $batt := batt - 1$
  else
    $gas := gas - 2;$
    $batt := batt + 1$
      fi;                 $\{miles + 1 \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$
    $miles := miles + 1$        $\{miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$
 od     $\{miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge gas \leq 1 \wedge batt \leq 0\}$
                                                    $\Rightarrow\{miles \geq gas_0 - 1\}$
```

Applying rule 8

Add postconditions to the branches of a conditional:
if e then S1 else S2 fi {Q}
becomes
if e then S1 {Q} else S2 {Q} fi {Q}

```
                                        $\{gas_0 = gas \wedge gas > 0 \wedge batt = 0\}$
$miles := \overline{0};$
{inv P $\triangleq$ $miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$
{dec $(gas + batt)\}$
while $gas > 1 \vee batt > 0$ do $\{miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge$
                                                    $gas > 1 \vee batt > 0\}$
```

```
  if $batt > 0$ then
    $batt := batt - 1$          $\{miles + 1 \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$
  else
    $gas := gas - 2;$
    $batt := batt + 1$          $\{miles + 1 \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$
      fi;                   $\{miles + 1 \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$
    $miles := miles + 1$        $\{miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$
 od     $\{miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge gas \leq 1 \wedge batt \leq 0\}$
                                                    $\Rightarrow\{miles \geq gas_0 - 1\}$
```

Here, we will use rule 1:

Prepend {wlp(x := e, Q)} to x := e {Q}.

```
                                            $\{gas_0 = gas \wedge gas > 0 \wedge batt = 0\}$
$miles := \overline{0};$
{inv P $\triangleq$ $miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$
{dec $(gas + batt)\}$
while $gas > 1 \vee batt > 0$ do $\{miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge gas > 1 \vee batt > 0\}$
  if $batt > 0$ then      $\{miles + 1 \geq gas_0 - (gas + batt - 1) \wedge gas \geq 0 \wedge batt - 1 \geq 0\}$
    $batt := batt - 1$          $\{miles + 1 \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$
  else               $\{miles + 1 \geq gas_0 - (gas - 2 + batt + 1) \wedge gas - 2 \geq 0 \wedge batt + 1 \geq 0\}$
     $gas := gas - 2;$ $\{miles + 1 \geq gas_0 - (gas + batt + 1) \wedge gas \geq 0 \wedge batt + 1 \geq 0\}$
     $batt := batt + 1$         $\{miles + 1 \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$
      fi;                   $\{miles + 1 \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$
    $miles := miles + 1$        $\{miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$
 od     $\{miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge gas \leq 1 \wedge batt \leq 0\}$
                                                    $\Rightarrow\{miles \geq gas_0 - 1\}$
```

Applying rule 3

Add preconditions to the branches of a conditional:
{P} if e then S1 else S2 fi
becomes
{P} if e then {P ∧ e} S1 else {P ∧ ¬e} S2 fi

```
                                   {$gas_0 = gas \wedge gas > 0 \wedge batt = 0$}
$miles := \bar{0}$;
{inv P ≜ $miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0$}
{dec $(gas + batt)$}
while $gas > 1 \vee batt > 0$ do {$miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge (gas > 1 \vee batt > 0)$}
  if $batt > 0$ then    {$miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge (gas > 1 \vee batt > 0) \wedge batt > 0$}
                        {$miles + 1 \geq gas_0 - (gas + batt - 1) \wedge gas \geq 0 \wedge batt - 1 \geq 0$}
     $batt := batt - 1$    {$miles + 1 \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0$}
  else        {$miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge (gas > 1 \vee batt > 0) \wedge batt \leq 0$}
          {$miles + 1 \geq gas_0 - (gas - 2 + batt + 1) \wedge gas - 2 \geq 0 \wedge batt + 1 \geq 0$}
      $gas := gas - 2$; {$miles + 1 \geq gas_0 - (gas + batt + 1) \wedge gas \geq 0 \wedge batt + 1 \geq 0$}
      $batt := batt + 1$    {$miles + 1 \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0$}
        fi;           {$miles + 1 \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0$}
     $miles := miles + 1$    {$miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0$}
 od    {$miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge gas \leq 1 \wedge batt \leq 0$}
                                                              ⇒{$miles \geq gas_0 - 1$}
```

Now, we have two proof obligations:

First,
{$miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge (gas > 1 \vee batt > 0) \wedge batt > 0$}
{$miles + 1 \geq gas_0 - (gas + batt - 1) \wedge gas \geq 0 \wedge batt - 1 \geq 0$}

{$miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge (gas > 1 \vee batt > 0) \wedge batt > 0$}
⇒{$miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge batt > 0 \wedge batt > 0$}
⇒{$miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge batt > 0$}
⇒{$miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 1$}
⇒{$miles + 1 \geq gas_0 - (gas + batt - 1) \wedge gas \geq 0 \wedge batt - 1 \geq 0$}
Second,
{$miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge (gas > 1 \vee batt > 0) \wedge batt \leq 0$}
{$miles + 1 \geq gas_0 - (gas - 2 + batt + 1) \wedge gas - 2 \geq 0 \wedge batt + 1 \geq 0$}

$\{miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0 \land (gas > 1 \lor batt > 0) \land batt \leq 0\}$
$\Rightarrow\{miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0 \land gas > 1 \land batt \leq 0\}$
$\Rightarrow\{miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land gas > 1 \land batt \geq 0 \land batt \leq 0\}$
$\Rightarrow\{miles \geq gas_0 - (gas + batt) \land gas > 1 \land batt = 0\}$
$\Rightarrow\{miles \geq gas_0 - (gas + batt) \land gas \geq 2 \land batt + 1 \geq 0\}$
$\Rightarrow\{miles + 1 \geq gas_0 - (gas - 2 + batt - 1) \land gas - 2 \geq 0 \land batt + 1 \geq 0\}$

We can rewrite our proof outline as:

```
                                        {$gas_0 = gas \land gas > 0 \land batt = 0$}
$miles := \bar{0}$;
{inv P ≜ $miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0$}
{dec $(gas + batt)$}
while $gas > 1 \lor batt > 0$ do {$miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0 \land (gas > 1 \lor batt > 0)$}
  if $batt > 0$ then     {$miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0 \land (gas > 1 \lor batt > 0) \land batt > 0$}
                         $\Rightarrow$ {$miles + 1 \geq gas_0 - (gas + batt - 1) \land gas \geq 0 \land batt - 1 \geq 0$}
     $batt := batt - 1$            {$miles + 1 \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0$}
  else            {$miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0 \land (gas > 1 \lor batt > 0) \land batt \leq 0$}
          $\Rightarrow$ {$miles + 1 \geq gas_0 - (gas - 2 + batt + 1) \land gas - 2 \geq 0 \land batt + 1 \geq 0$}
       $gas := gas - 2$; {$miles + 1 \geq gas_0 - (gas + batt + 1) \land gas \geq 0 \land batt + 1 \geq 0$}
       $batt := batt + 1$          {$miles + 1 \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0$}
        fi;                  {$miles + 1 \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0$}
     $miles := miles + 1$          {$miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0$}
 od      {$miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0 \land gas \leq 1 \land batt \leq 0$}
                                                            $\Rightarrow$ {$miles \geq gas_0 - 1$}
```

Propagating our loop invariant back and applying rule 1:

Prepend {wlp(x := e, Q)} to x := e {Q}.

```
                                {$gas_0 = gas \land gas > 0 \land batt = 0$}
                        {$0 \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0$}
$miles := \bar{0}$;             {$miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0$}
{inv P ≜ $miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0$}
{dec $(gas + batt)$}
```

```
while $gas > 1 \lor batt > 0$ do {$miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0 \land$
                                                        $(gas > 1 \lor batt > 0)$}
  if $batt > 0$ then      {$miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0 \land (gas >$
                    $1 \lor batt > 0) \land batt > 0$}
                      $\Rightarrow${$miles + 1 \geq gas_0 - (gas + batt - 1) \land gas \geq 0 \land batt - 1 \geq 0$}
    $batt := batt - 1$            {$miles + 1 \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq$
                                                                    $0$}
  else              {$miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0 \land (gas > 1 \lor$
        $batt > 0) \land batt \leq 0$}
        $\Rightarrow${$miles + 1 \geq gas_0 - (gas - 2 + batt + 1) \land gas - 2 \geq 0 \land batt + 1 \geq 0$}
     $gas := gas - 2$; {$miles + 1 \geq gas_0 - (gas + batt + 1) \land gas \geq 0 \land batt + 1 \geq$
                                                                    $0$}
     $batt := batt + 1$           {$miles + 1 \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq$
                                                                    $0$}
        fi;                    {$miles + 1 \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0$}
    $miles := miles + 1$          {$miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0$}
  od       {$miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0 \land gas \leq 1 \land batt \leq 0$}
                                                        $\Rightarrow${$miles \geq gas_0 - 1$}
```

We have one more proof obligation,

{$gas_0 = gas \land gas > 0 \land batt = 0$}
{$0 \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0$}

{$gas_0 = gas \land gas > 0 \land batt = 0$}
$\Rightarrow${$gas_0 - gas = 0 \land gas > 0 \land batt = 0$}
$\Rightarrow${$gas_0 - gas + 0 = 0 \land gas > 0 \land batt = 0$}
$\Rightarrow${$gas_0 - gas + batt = 0 \land gas > 0 \land batt = 0$}
$\Rightarrow${$gas_0 - (gas + batt) \leq 0 \land gas \geq 0 \land batt \geq 0$}
$\Rightarrow${$0 \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0$}

Full proof outline for partial correctness:

```
                                 {$gas_0 = gas \land gas > 0 \land batt = 0$}
                       $\Rightarrow${$0 \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0$}
$miles := \bar{0}$;              {$miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0$}
{inv P $\triangleq$ $miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0$}
{dec $(gas + batt)$}
while $gas > 1 \lor batt > 0$ do {$miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0 \land$
                                                        $(gas > 1 \lor batt > 0)$}
  if $batt > 0$ then      {$miles \geq gas_0 - (gas + batt) \land gas \geq 0 \land batt \geq 0 \land (gas >$
                    $1 \lor batt > 0) \land batt > 0$}
                      $\Rightarrow${$miles + 1 \geq gas_0 - (gas + batt - 1) \land gas \geq 0 \land batt - 1 \geq 0$}
```

$batt := batt - 1$ $\{miles + 1 \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$

else $\{miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge (gas > 1 \vee batt > 0) \wedge batt \leq 0\}$

$\Rightarrow \{miles + 1 \geq gas_0 - (gas - 2 + batt + 1) \wedge gas - 2 \geq 0 \wedge batt + 1 \geq 0\}$

$gas := gas - 2;$ $\{miles + 1 \geq gas_0 - (gas + batt + 1) \wedge gas \geq 0 \wedge batt + 1 \geq 0\}$

$batt := batt + 1$ $\{miles + 1 \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$

fi; $\{miles + 1 \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$

$miles := miles + 1$ $\{miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$

od $\{miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge gas \leq 1 \wedge batt \leq 0\}$

$\Rightarrow \{miles \geq gas_0 - 1\}$

Proving Termination:

For termination, we have to prove the additional proof obligations for decreasing clause:

1) P ⇒ t ≥ 0
2) {P ∧ e ∧ t = t_0} S {P ∧ t < t_0}

$\{gas_0 = gas \wedge gas > 0 \wedge batt = 0\}$

$\Rightarrow \{0 \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$

$miles := \bar{0};$ $\{miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0\}$

{inv P ≜ $miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0$}

{dec $(gas + batt)$}

while $gas > 1 \vee batt > 0$ do $\{miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge (gas > 1 \vee batt > 0) \wedge (gas + batt = t_0)\}$

if $batt > 0$ then $\{miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge (gas > 1 \vee batt > 0) \wedge batt > 0 \wedge (gas + batt = t_0)\}$

$\Rightarrow \{miles + 1 \geq gas_0 - (gas + batt - 1) \wedge gas \geq 0 \wedge batt - 1 \geq 0 \wedge (gas + batt - 1 < t_0)\}$

$batt := batt - 1$ $\{miles + 1 \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge (gas + batt < t_0)\}$

else $\{miles \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge (gas > 1 \vee batt > 0) \wedge batt \leq 0 \wedge (gas + batt = t_0)\}$

$\Rightarrow \{miles + 1 \geq gas_0 - (gas - 2 + batt + 1) \wedge gas - 2 \geq 0 \wedge batt + 1 \geq 0 \wedge (gas - 2 + batt + 1 < t_0)\}$

$gas := gas - 2;$ $\{miles + 1 \geq gas_0 - (gas + batt + 1) \wedge gas \geq 0 \wedge batt + 1 \geq 0 \wedge (gas + batt + 1 < t_0)\}$

$batt := batt + 1$ $\{miles + 1 \geq gas_0 - (gas + batt) \wedge gas \geq 0 \wedge batt \geq 0 \wedge (gas + batt < t_0)\}$

```
    fi;                    ${miles + 1 \geq\ gas_0 - (gas + batt) \wedge gas \geq 0\ \wedge batt \geq 0 \wedge$
                                                               $(gas + batt < t_0)}$
  $miles := miles + 1$         ${miles \geq\ gas_0 - (gas + batt) \wedge gas \geq 0\ \wedge batt \geq 0 \wedge$
                                                               $(gas + batt < t_0)}$
od      ${miles \geq\ gas_0 - (gas + batt) \wedge gas \geq 0\ \wedge batt \geq 0 \wedge gas \leq 1\ \wedge batt \leq 0}$
                                                        $\Longrightarrow{miles \geq\ gas_0 - 1}$
```

Additional Proof obligations:

1) P ⇒ t ≥ 0

   $\{miles \geq\ gas_0 - (gas + batt) \wedge gas \geq 0\ \wedge batt \geq 0\}$
   $\Longrightarrow\{(gas + batt) \geq 0\}$

   This is clear as $gas \geq 0\ \wedge batt \geq 0$ so $gas + batt \geq 0$

2) $\{miles \geq\ gas_0 - (gas + batt) \wedge gas \geq 0\ \wedge batt \geq 0 \wedge (gas > 1\ \vee batt > 0)\ \wedge batt > 0 \wedge (gas + batt = t_0)\}$
   $\Longrightarrow\{miles + 1 \geq gas_0 - (gas + batt - 1) \wedge gas \geq 0 \wedge batt - 1 \geq 0 \wedge (gas + batt - 1 < t_0)\}$

   This is clear as $(gas + batt - 1 < t_0) \Longrightarrow (gas + batt = t_0)$

3) $\{miles \geq\ gas_0 - (gas + batt) \wedge gas \geq 0\ \wedge batt \geq 0 \wedge (gas > 1\ \vee batt > 0)\ \wedge batt \leq 0 \wedge (gas + batt = t_0)\}$
   $\Longrightarrow\{miles + 1 \geq gas_0 - (gas - 2 + batt + 1) \wedge gas - 2 \geq 0 \wedge batt + 1 \geq 0 \wedge (gas - 2 + batt + 1 < t_0)\}$

   This is clear as $(gas - 2 + batt + 1 < t_0) \Longrightarrow (gas + batt - 1 < t_0)$
   $\Longrightarrow (gas + batt = t_0)$

This proves termination.

As we proved termination as well as partial correctness. So, we can say,

```
                                         $[gas_0 = gas\ \wedge gas > 0\ \wedge batt = 0]$
$miles := \bar{0};$
{inv $miles \geq\ gas_0 - (gas + batt) \geq \wedge gas \geq 0\ \wedge batt \geq 0$}
{dec $(gas + batt)$}
while $gas > 1\ \vee batt > 0$ do
  if $batt > 0$ then
    $batt := batt - 1$
  else
```

```
      $gas := gas - 2$;
      $batt := batt + 1$
   fi;
   $miles := miles + 1$
od
                                    $[miles \geq gas_0 - 1]$
```

This can be verified from the Dafny outputs as well.

Task 1.2

Given Program:

```
                                    $[\forall i \in \mathbb{Z}. (0 \leq i < |a|) \rightarrow a[i] \geq 0]$
i := $\bar{0}$;
{inv ____}
{dec ____}
while i < size(a) do
   {inv ____}
   {dec ____}
   while a[i] > 0 do
      a[i] := a[i] – 1
   od;
   i := i + 1
od
                                    $[\forall i \in \mathbb{Z}. (0 \leq i < |a|) \rightarrow a[i] = 0]$
```

Ans.

Here we must provide an invariant and a loop bound for each while loop in this program. As the program has nested loops, we need two invariants and two loop bounds.

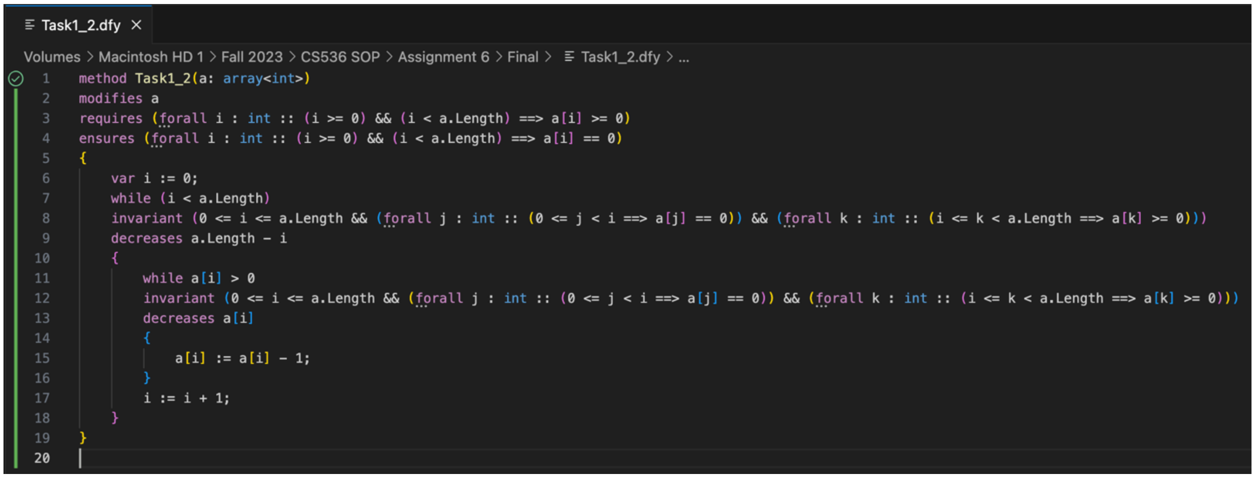
```
                                    $[\forall i \in \mathbb{Z}. (0 \leq i < |a|) \rightarrow a[i] \geq 0]$
i := $\bar{0}$;
{inv $0 \leq i \leq |a| \wedge \forall j \in \mathbb{Z}. (0 \leq j < i) \rightarrow a[j] = 0 \wedge \forall k \in \mathbb{Z}. (i \leq k < |a|) \rightarrow a[k] \geq 0$}
{dec |a| – i}
while i < size(a) do
   {inv $0 \leq i \leq |a| \wedge \forall j \in \mathbb{Z}. (0 \leq j < i) \rightarrow a[j] = 0 \wedge \forall k \in \mathbb{Z}. (i \leq k < |a|) \rightarrow a[k] \geq 0$}
   {dec a[i]}
   while a[i] > 0 do
      a[i] := a[i] – 1
   od;
   i := i + 1
```

od

$$[\forall i \ \in \ \mathbb{Z}. (0 \leq i < |a|) \rightarrow a[i] = 0]$$

Understanding of the program:
We are iterating through an array, and we are decreasing every element to 0.

Coming up with the invariant:
I took the approach that at point of time in the array, #j elements will be reduced to 0 and #k of original elements will be 0 or greater than 0.
These invariants and loop bound work as can be seen in the below snip of Dafny.



Task1_2.dfy ×
Volumes > Macintosh HD 1 > Fall 2023 > CS536 SOP > Assignment 6 > Final > Task1_2.dfy > ...

```
method Task1_2(a: array<int>)
modifies a
requires (forall i : int :: (i >= 0) && (i < a.Length) ==> a[i] >= 0)
ensures (forall i : int :: (i >= 0) && (i < a.Length) ==> a[i] == 0)
{
    var i := 0;
    while (i < a.Length)
    invariant (0 <= i <= a.Length && (forall j : int :: (0 <= j < i ==> a[j] == 0)) && (forall k : int :: (i <= k < a.Length ==> a[k] >= 0)))
    decreases a.Length - i
    {
        while a[i] > 0
        invariant (0 <= i <= a.Length && (forall j : int :: (0 <= j < i ==> a[j] == 0)) && (forall k : int :: (i <= k < a.Length ==> a[k] >= 0)))
        decreases a[i]
        {
            a[i] := a[i] - 1;
        }
        i := i + 1;
    }
}
```

For each bound expression:

1. Why the loop invariant of the same loop implies the bound expression is always nonnegative.

Answer:
For the first bound expression {dec |a| – i}, the invariant {inv $0 \leq i \leq |a| \wedge \forall j \in \mathbb{Z}. (0 \leq j < i) \rightarrow a[j] = 0 \wedge \forall k \in \mathbb{Z}. (i \leq k < |a|) \rightarrow a[k] \geq 0$} clearly says that:
$i \leq |a|$
$\Longrightarrow 0 \leq |a| - i$
$\Longrightarrow |a| - i \geq 0$
This is verified by the Dafny output.

For the second bound expression {dec a[i]}, the invariant {inv $0 \leq i \leq |a| \wedge \forall j \in \mathbb{Z}. (0 \leq j < i) \rightarrow a[j] = 0 \wedge \forall k \in \mathbb{Z}. (i \leq k < |a|) \rightarrow a[k] \geq 0$} clearly says that:
First j elements are 0 and next k elements are either 0 or greater than 0.
$\Longrightarrow a[i] \geq 0$
This is verified by the Dafny output.

2. How you know the bound expression decreases each loop iteration.

Answer:
For the first bound expression {dec |a| – i}, the loop sets i := i + 1, the first intuition might be to use t = −i. However, this makes it even more likely that t can be negative, so we need to add something to the bound. We know that i never goes above |a|, so we use |a| − i. This is verified by the Dafny output.

For the second bound expression {dec a[i]}, the loop sets a[i] := a[i] - 1, the first intuition might be to use t = a[i] as it is decreasing every iteration as per the program working. We know that a[i] never goes below 0 from our discussion in Q1 of Task 1.2, so a[i] is a valid bound expression. This is verified by the Dafny output.

Task 1.3

a) Given t is a valid bound expression, i is a variable, and k is a constant.
   To check: $\sqrt{t}$ is a valid bound expression or not.

Ans.
Even if t is a valid bound expression, it might happen that $\sqrt{t}$ can result into a non-integer value.
By definition, square root of a number can be negative as well. (t = $\pm\sqrt{t}$)
The rules of bound expressions say that the value of bound expression should be non-negative integer.
So, $\sqrt{t}$ is not a valid bound expression.
This is my understanding.

TO THE TA/PROFESSOR GRADING THIS: The notes for loop bound mention a fact that if t is a valid bound expression, so is a.t^n + b for any positive values of a, b, and n.
Here the fact is a bit ambiguous, as it does not mention that values of a, b, and n should always be integers or can be non-integer values only.
Because one can consider a case where a=1, b=0, and n=1/2 which exactly says $\sqrt{t}$.
Here a, b, and n are all positive, but n is a decimal fraction and not an integer.
As per this rule, if t is a valid bound expression, 1.$t^{(1/2)}$ + 0 is also a valid bound expression.

The confusion here is, if $\sqrt{t}$ is a non-integer number a loop cannot run for a non-integer number of times. Hence, if the fact mentioned in the notes works only for integer values, it is not a valid bound expression. If it applies to any positive value of a, b, and n which can be non-integer values as well. The $\sqrt{t}$ is also a valid bound expression as per the fact mentioned in the notes.

b) Given t is a valid bound expression, i is a variable, and k is a constant.
To check: t^2 is a valid bound expression or not.

Ans.
t is a valid bound expression.
By definition, square of a number is always positive. $(\pm t)^2 = t^2$
The rules of bound expressions say that the value of bound expression should be non-negative and t^2 satisfies that rule.
So, t^2 is a valid bound expression.

c) Given t is a valid bound expression, i is a variable, and k is a constant.
To check: $t + i$ is a valid bound expression or not.

Ans.
t is a valid bound expression. Assumption: i is in the loop.
The rules of bound expressions say that the value of bound expression should always be non-negative and $t + i$ may or may not result into a non-negative number which is not desirable. As the value of i dominates the overall value of $t + i$, we cannot say that $t + i$ will always be non-negative.
So, $t + i$ is not a valid bound expression.

d) Given t is a valid bound expression, i is a variable, and k is a constant.
To check: $t + i^2$ is a valid bound expression or not.

Ans.
t is a valid bound expression. Assumption: i is in the loop.
The rules of bound expressions say that the value of bound expression should always be non-negative and $t + i^2$ may or may not result into a non-negative number which is not desirable. As the value of i (in turn, i^2) dominates the overall value of $t + i^2$, we cannot say that $t + i^2$ will always be non-negative.
So, $t + i^2$ is not a valid bound expression.

e) Given t is a valid bound expression, i is a variable, and k is a constant.
To check: $t + k$ is a valid bound expression or not.

Ans.
t is a valid bound expression. Assumption: k is in the loop.
The rules of bound expressions say that the value of bound expression should always be non-negative and $t + k$ may or may not result into a non-negative number which is not desirable. As the value of k dominates the overall value of $t + k$, we cannot say that $t + k$ will always be non-negative.
So, $t + k$ is not a valid bound expression.

f) Given t is a valid bound expression, i is a variable, and k is a constant.
To check: $t + k^2$ is a valid bound expression or not.

Ans.
t is a valid bound expression. Assumption: k is in the loop.
The rules of bound expressions say that the value of bound expression should always be non-negative and $t + k^2$ will always result into a non-negative number. As the value of k^2 is a positive constant value and the overall value of $t + k^2$ will always reduce, so we can say that $t + k^2$ will always be non-negative.
So, $t + k^2$ is a valid bound expression.

2. Weakest Preconditions with Array Assignments.

Task 2.1

a) wlp(a[if x = $\overline{0}$ then i else j] := 1, a[i] = 1)

Ans.

wlp(a[if x = $\overline{0}$ then i else j] := 1, a[i] = 1)
::=[1/a[if x = $\overline{0}$ then i else j]]a[i] = 1
::=(if i = (if x = $\overline{0}$ then i else j) then 1 else a[i]) = 1
::=(if i = (if x = $\overline{0}$ then i else j) then 1=1 else a[i]) = 1
::=(if i = (if x = $\overline{0}$ then i else j) then T else a[i]) = 1
⟺ i = (if x = $\overline{0}$ then i else j) ∨ a[i] = 1
⟺ (if x = $\overline{0}$ then i=i else i=j) ∨ a[i] = 1
::=(if x = $\overline{0}$ then T else i=j) ∨ a[i] = 1
⟺ (x = $\overline{0}$ ∨ i=j) ∨ a[i] = 1

b) wlp(a[i] := $\overline{5}$, a[a[1]] = 5)

Ans.

wlp(a[i] := $\overline{5}$, a[a[1]] = 5)
::= [$\overline{5}$/a[i]](a[a[1]] = 5)
::= [$\overline{5}$/a[i]](a[a[1]]) = [$\overline{5}$/a[i]](5)
::= (if(if 1=i then 5 else a[1]) = i then 5 else a[if 1=i then 5 else a[1]])) = 5
::= if(if 1=i then 5 else a[1]) = i then 5 = 5 else a[if 1=i then 5 else a[1]) = 5
::= if(if 1=i then 5 else a[1]) = i then T else a[if 1=i then 5 else a[1]) = 5
⟺ ((if 1=i then 5 else a[1]) = i) ∨ (a[if 1=i then 5 else a[1]) = 5)
::= (if 1=i then 5=i else a[1]=i) ∨ (a[if 1=i then 5 else a[1]) = 5)
⟺ ((1=i ∧ 5=i) ∨ (1≠i ∧ a[1]=i)) ∨ (a[if 1=i then 5 else a[1]) = 5)
⇒ ((1=i ∧ 5=i) ∨ (1≠i ∧ a[1]=i)) ∨ (if 1=i then a[5] else a[a[1]]) = 5)

$::= ((1=i \land 5=i) \lor (1\neq i \land a[1]=i)) \lor (\text{if } 1=i \text{ then } a[5]=5 \text{ else } a[a[1]] = 5)$

$\Leftrightarrow ((1=i \land 5=i) \lor (1\neq i \land a[1]=i)) \lor ((1=i \land a[5]=5) \lor (1\neq i \land a[a[1]] = 5))$

$::= (F) \lor (1\neq i \land a[1]=i) \lor (1=i \land a[5]=5) \lor (1\neq i \land a[a[1]] = 5)$

$\Leftrightarrow (1\neq i \land a[1]=i) \lor (1=i \land a[5]=5) \lor (1\neq i \land a[a[1]] = 5)$ [By Identity law $p \lor F \Leftrightarrow p$]

$\Leftrightarrow (1\neq i \land a[1]=i) \lor (1\neq i \land a[a[1]] = 5) \lor (1=i \land a[5]=5)$

[By Commutative law $p \lor q \Leftrightarrow q \lor p$]

$\Leftrightarrow (1\neq i \land (a[1]=i \lor a[a[1]] = 5)) \lor (1=i \land a[5]=5)$

[By Distributive law $(p \land q) \lor (p \land r) \Leftrightarrow p \land (q \lor r)$]

c) $wlp(a[j] := a[i] + \overline{1}, a[j] > a[i])$

Ans.

$wlp(a[j] := a[i] + \overline{1}, a[j] > a[i])$

$::= [(a[i] + \overline{1})/a[j]](a[j] > a[i])$

$::= [(a[i] + \overline{1})/a[j]](a[j]) > [(a[i] + \overline{1})/a[j]](a[i])$

$::= (\text{if } j=j \text{ then } a[i] + \overline{1} \text{ else } a[j]) > (\text{if } i=j \text{ then } a[i] + \overline{1} \text{ else } a[i])$

$\Rightarrow a[i] + \overline{1} > (\text{if } i=j \text{ then } a[i] + \overline{1} \text{ else } a[i])$

$\Leftrightarrow (\text{if } i=j \text{ then } a[i] + \overline{1} > a[i] + \overline{1} \text{ else } a[i] + \overline{1} > a[i])$

$\Leftrightarrow (\text{if } i=j \text{ then } F \text{ else } a[i] + \overline{1} > a[i])$

$\Leftrightarrow i\neq j \land (a[i] + \overline{1} > a[i])$

$\Leftrightarrow i\neq j \land (T)$ [Similar logic: $x + 1 > x$]

$\Leftrightarrow i\neq j$ By Identity law, $p \land T \Leftrightarrow p$

d) $wlp(i=5; a[i] := a[i+1], a[i] > 0)$

Ans.

$wlp(i=5; a[i] := a[i+1], a[i] > 0)$

$::=wlp(i=5, wlp(a[i] := a[i+1], a[i] > 0))$

For this we need to compute $wlp(a[i] := a[i+1], a[i] > 0)$.

$wlp(a[i] := a[i+1], a[i] > 0)$

$::= [a[i+1]/a[i]](a[i] > 0)$

$::= [a[i+1]/a[i]](a[i]) > [a[i+1]/a[i]](0)$

$::= [a[i+1]/a[i]](a[i]) > 0$

$::= (\text{if } i=i \text{ then } a[i+1] \text{ else } a[i]) > 0$

$::= a[i+1] > 0$

Substituting $wlp(a[i] := a[i+1], a[i] > 0) = (a[i+1]>0)$

$wlp(i=5, a[i+1] > 0)$

$::= [5/i]\ (a[i+1] > 0)$

::= a[5+1] > 0
::= a[6] > 0

e) wp(i=5; a[i] := a[i+1], a[i] > 0)

Ans.
wp(i=5; a[i] := a[i+1], a[i] > 0)
$::= wlp(i=5; a[i] := a[i+1], a[i] > 0) \land D(i=5; a[i] := a[i+1])$

From Task 2.1 (e), we can substitute wlp(i=5; a[i] := a[i+1], a[i] > 0) = a[6] > 0

For this, we need to compute: D(i=5; a[i] := a[i+1])

D(i=5; a[i] := a[i+1])
$::= D(i:=5) \land wlp(i:=5, D(a[i] := a[i+1]))$

For this, we need to compute: D(i:=5)

D(i:=5)
::= D(5)
::= T

Also, we need to compute: wlp(i:=5, D(a[i] := a[i+1]))

However, we need D(a[i] := a[i+1]) to solve first.

By definition, $D(a[e_1] := e_2) ::= D(e_1) \land D(e_2) \land 0 \leq e_1 < |a|$
D(a[i] := a[i+1])
$::= D(i) \land D(a[i+1]) \land 0 \leq i < |a|$
$::= T \land D(i+1) \land 0 \leq i+1 < |a| \land 0 \leq i < |a|$
$::= T \land D(i) \land D(1) \land 0 \leq i+1 < |a| \land 0 \leq i < |a|$
$::= T \land T \land T \land 0 \leq i+1 < |a| \land 0 \leq i < |a|$
$::= 0 \leq i+1 < |a| \land 0 \leq i < |a|$

wlp(i:=5, D(a[i] := a[i+1]))
$::= wlp(i:=5, 0 \leq i+1 < |a| \land 0 \leq i < |a|)$
$::= [5/i](0 \leq i+1 < |a| \land 0 \leq i < |a|)$
$::= [5/i](0 \leq i+1 < |a|) \land [5/i](0 \leq i < |a|)$
$::= (0 \leq 5+1 < |a| \land 0 \leq 5 < |a|)$
$::= (0 \leq 6 < |a| \land 0 \leq 5 < |a|)$

Re-substituting everything in wp formula above.
wp(i=5; a[i] := a[i+1], a[i] > 0)

$::= a[6] > 0 \land T \land (0 \leq 6 < |a| \land 0 \leq 5 < |a|)$

$::= a[6] > 0 \land (0 \leq 6 < |a| \land 0 \leq 5 < |a|)$

f) $wlp(\text{if } i=j \text{ then } j := j+1 \text{ else } a[j] := a[i]+1 \text{ fi}, a[j] > a[i])$

Ans.

$wlp(\text{if } i=j \text{ then } j := j+1 \text{ else } a[j] := a[i]+1 \text{ fi}, a[j] > a[i])$

$::= (i=j \rightarrow wlp(j := j+1, a[j] > a[i])) \land (i\neq j \rightarrow wlp(a[j] := a[i]+1, a[j] > a[i]))$

$wlp(j := j+1, a[j] > a[i])$

$::= [j+1/j](a[j] > a[i])$

$::= [j+1/j](a[j]) > [j+1/j](a[i])$

$::= a[j+1] > a[i]$

$wlp(a[j] := a[i]+1, a[j] > a[i])$

$::= [(a[i]+1) / a[j]](a[j] > a[i])$

$::= [(a[i]+1) / a[j]](a[j]) > [(a[i]+1) / a[j]](a[i])$

$::= (\text{if } j=j \text{ then } a[i] + 1 \text{ else } a[j]) > (\text{if } i=j \text{ then } a[i]+1 \text{ else } a[i])$

$\Rrightarrow (a[i] + 1) > (\text{if } i=j \text{ then } a[i]+1 \text{ else } a[i])$

$\Leftrightarrow (\text{if } i=j \text{ then } a[i] + \overline{1} > a[i] + 1 \text{ else } a[i] + 1 > a[i])$

$\Leftrightarrow (\text{if } i=j \text{ then } F \text{ else } a[i] + 1 > a[i])$

$\Leftrightarrow i\neq j \land (a[i] + 1 > a[i])$

$\Leftrightarrow i\neq j \land (T)$ [Similar logic: $x + 1 > x$]

$\Leftrightarrow i\neq j$ [By Identity law, $p \land T \Leftrightarrow p$]

Substituting back into the above formula,

$wlp(\text{if } i=j \text{ then } j := j+1 \text{ else } a[j] := a[i]+1 \text{ fi}, a[j] > a[i])$

$::= (i=j \rightarrow wlp(j := j+1, a[j] > a[i])) \land (i\neq j \rightarrow wlp(a[j] := a[i]+1, a[j] > a[i]))$

$::= (i=j \rightarrow a[j+1] > a[i]) \land (i\neq j \rightarrow i\neq j)$

$::= (i=j \rightarrow a[j+1] > a[i]) \land T$

$\Leftrightarrow (i=j \rightarrow a[j+1] > a[i])$ [By Identity law, $p \land T \Leftrightarrow p$]

$\Leftrightarrow (\neg i=j \lor a[j+1] > a[i])$ [By definition of conditional, $p \rightarrow q \Leftrightarrow \neg p \lor q$]

$\Leftrightarrow (i\neq j \lor a[j+1] > a[i])$

3. One more wrap-up question

Task 3.1

How long (approximately) did you spend on this homework, in total hours of actual working time?

Ans. Totally I spent 22 hours on this assignment.