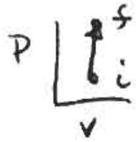
SOLUTIONS TO UNIFIED T2 (WAITZ)

a) GIVEN TO PROPERTIES FOR INITIAL COND. THEREFORE STATE IS DEFINED.

$$v_i = \frac{RT_i}{P_i} \qquad R_{He} = \frac{\mathcal{R}}{MW} = \frac{8.314}{4} = 2.078 \frac{kJ}{kg\text{-}K}$$

$$v_i = \frac{(2.078\times 10^3)(300)}{100\times 10^6} = 0.0062 \frac{m^3}{kg}$$

GIVEN PATH TO FINAL STATE (RIGID TANK $\Rightarrow \Delta v = 0$)



AND GIVEN FINAL TEMPERATURE.

SO $v_f = v_i = 0.0062$, $T_f = 400K$

$$\therefore P_f = \frac{RT_f}{v_f} = \frac{(2078)400}{0.0062} = 133 MPa$$

b) **NO WORK WAS DONE.** $\int p\,dv = 0$ SINCE $dv = 0$

c) HERE A NEW PATH IS SPECIFIED, INSTEAD OF V= CONST, WE HAVE $\frac{dp}{dv} = 1\times 10^5 \frac{MPa}{m^3/kg}$ $\therefore p = 1\times 10^5 v + Const.$

SOLVE FOR CONSTANT USING P_i, v_i, YOU GET

$$p = 1\times 10^5 \frac{MPa}{m^3/kg} \cdot v - 523.4\ MPa$$

} NEW PATH FULLY DEFINED

BUT NOT GIVEN P_f OR v_f, INSTEAD WE ARE GIVEN T_f SO SUBSTITUTE, USING IDEAL GAS

$$p = \frac{RT_f}{v_f} \quad \text{(UNITS Pa)}$$

$$\frac{2078\cdot 400}{v_f} = 1\times 10^{11} v_f - 523.4\times 10^6$$

(NOTE EXTRA 10^6 TO MAKE UNITS CONSISTENT)

REWRITING, $v_f^2 - 5.234\times10^{-3}\, v_f - 8.314\times10^{-6} = 0$

BY QUADRATIC FORMULA $v_f = \dfrac{5.234\times10^{-3} \pm 7.788\times10^{-3}}{2}$

$$\boxed{v_f = 0.00651\ \frac{m^3}{kg}}$$

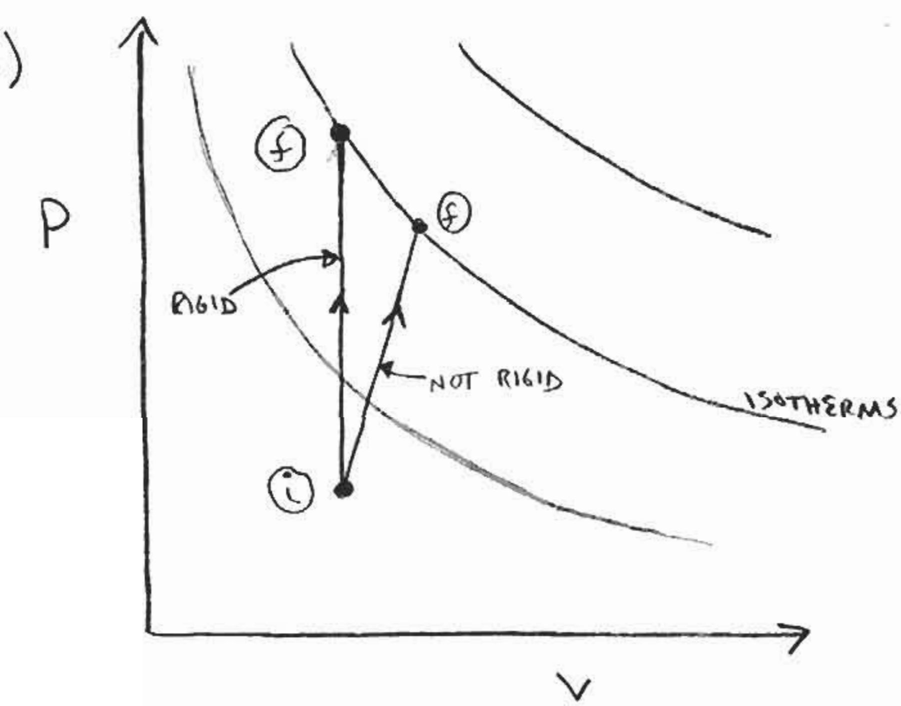
$$P_f = \frac{RT_f}{v_f} = \frac{2078.400}{0.00651} = \boxed{127.7\ MPa}$$

d) $$w = \int_{v_1}^{v_2} p\,dv = \int_{v_1}^{v_2} \left(1\times10^5 \frac{MPa}{m^3/kg}\cdot v - 523.4\ MPa\right) dv$$

$$w = \frac{1\times10^{11}}{2} v^2 - 523.4\times10^6\, v \Bigg|_{v_1}^{v_2}$$

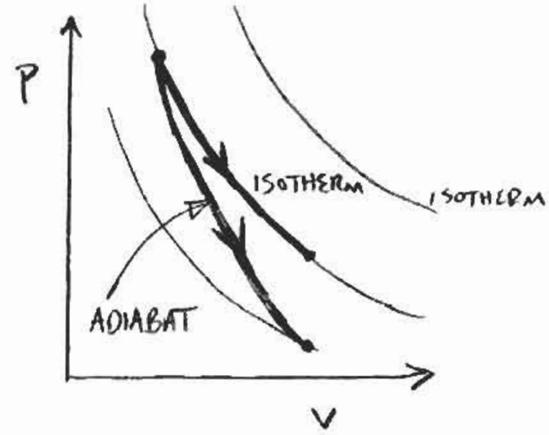
$$\boxed{w = 31.4\ kJ/kg}$$

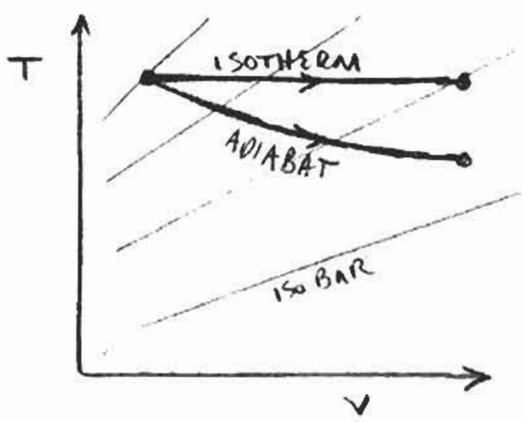
e)



SOLUTIONS TO UNIFIED T3 (WAITZ)

a)





b) Q-S ISOTHERMAL

$T_1 = 300K$, $v_1 = 1\ m^3/kg$

$T_2 = 300K$, $v_2 = 10\ m^3/kg$

$\therefore\ P_2 = \frac{287 \cdot 300}{10}$

$\boxed{P_2 = 8610\ Pa}$

c) $w = RT \ln\left(\frac{v_2}{v_1}\right)$

$= 287 \cdot 300 \cdot \ln(10)$

$\boxed{w = 198\ kJ/kg}$

$\Delta u = q - w = C_v \Delta T = 0$

$\therefore\ \boxed{q = w = 198\ kJ/kg}$

Q-S ADIABATIC

$pv^{\gamma} = \text{CONST.}$ $\gamma = 1.4$

$P_1 = \frac{287 \cdot 300}{1} = 86100\ Pa$

$\frac{P_2}{P_1} = \left(\frac{v_1}{v_2}\right)^{\gamma}$ $\therefore$

$\boxed{P_2 = 3428\ Pa}$ $T_2 = \frac{P_2 v_2}{R}$

$\boxed{T_2 = 119\ K}$

$\Delta u = \cancel{q}^{\,0} - w$

$\boxed{q = 0}$

$w = -\Delta u = -C_v(T_2 - T_1)$

$= -716.5(119 - 300)$

$\boxed{w = 129\ kJ/kg}$

q-S ISOTHERMAL	q-S ADIABATIC
d) $h = u + pv$	$h = u + pv$
$dh = C_p dT$	$dh = C_p dT$
$\Delta h = 1003.5 (T_2 - T_1)$	$\Delta h = 1003.5 (T_2 - T_1)$
$\Delta h = 0$	$= 1003.5 (119 - 300)$
	$\boxed{\Delta h = -181.6 \text{ kJ}}$

e) HEAT IS A TRANSFER OF ENERGY ACROSS A SYSTEM BOUNDARY BY VIRTUE OF A TEMPERATURE DIFFERENCE ONLY. IT IS MEASURED IN JOULES

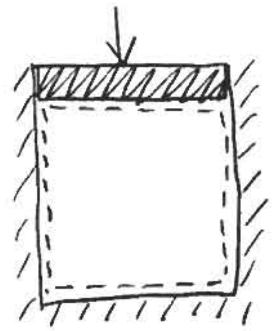
TEMPERATURE IS A THERMODYNAMIC PROPERTY AND A FUNCTION OF THE STATE OF A SYSTEM. IT IS MEASURED IN KELVIN.

* IT IS POSSIBLE TO HAVE AN ISOTHERMAL PROCESS WITH HEAT TRANSFER
* IT IS POSSIBLE TO HAVE AN ADIABATIC PROCESS WITH A TEMPERATURE CHANGE

AS DEMONSTRATED IN THIS PROBLEM

SOLUTIONS TO T4 (WAITZ)

a) QUASI-STATIC, ADIABATIC COMPRESSION



$p_1 = 100\,kPa$, $p_2 = 500\,kPa$

$T_1 = 300\,K$

$$V_1 = \frac{287 \cdot 300}{100{,}000} = 0.861\ \frac{m^3}{kg}$$

$pV^{\gamma} = CONST.$ (FOR WHOLE PROCESS) $\therefore \frac{p_2}{p_1} = \left(\frac{V_1}{V_2}\right)^{\gamma}$ $\quad 5 = \left(\frac{0.861}{V_2}\right)^{1.4}$

$\therefore$ $\boxed{V_2 = 0.273}$

$$T_2 = \frac{500{,}000 \cdot 0.273}{287} = \boxed{475\,K}$$

$\Delta u = \cancel{q}^{\,0} - w \quad \Delta u = -w, \; C_v \Delta T = -w$

$w = -716.5(475-300) = \boxed{-125\,kJ/kg}$

BY THE SYSTEM — NEGATIVE SINCE ENERGY TRANSFERRED TO SYSTEM.

b) INITIAL STATE IS THE SAME.
KNOW $p_f = 500\,kPa$ BUT DON'T KNOW T_f OR V_f

* DON'T KNOW BEHAVIOR OF STATE OF SYSTEM DURING PROCESS ⟶ BUT FIRST LAW STILL HOLDS.

* HOWEVER, NOW WE MUST USE EXTERNAL INFORMATION TO RELATE PROPERTIES AT INITIAL AND FINAL STATE

1ST LAW: $\Delta u = \cancel{q}^{0} - w$

(STILL INSULATED)

$$C_v \Delta T = -w = -P_{ext} \Delta V$$

1) $C_v(T_f - T_i) = -P_{ext}(V_f - V_i)$

AND IDEAL GAS LAW HOLDS AT INITIAL CONDITION AND FINAL CONDITION (BUT NOT IN BETWEE)

2) $P_f V_f = R T_f$ WHERE $P_f = P_{ext}$

2 EQNS IN 2 UNKNOWNS (T_f, V_f)

SUBSTITUTING,

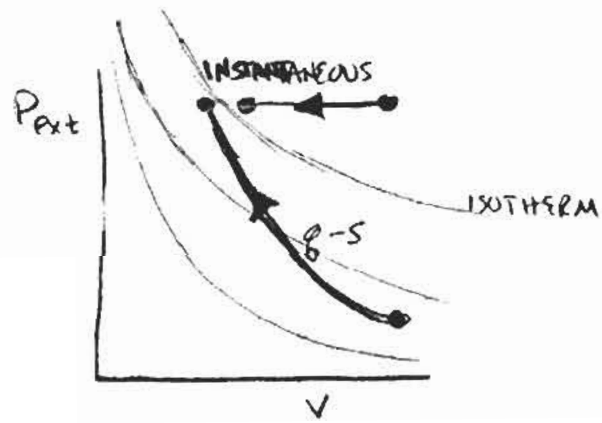
$$T_f = \frac{P_{ext} V_i + C_v T_i}{C_v + R} = \boxed{643.2\ K}$$

$$V_f = \frac{287 \cdot 643.2}{500{,}000} = \boxed{0.369\ m^3/kg}$$

ENDED UP AT A DIFFERENT STATE!

$$W = -C_v \Delta T = -716.5(643.2 - 300) = \boxed{-245.9\ kJ/kg}$$

c)



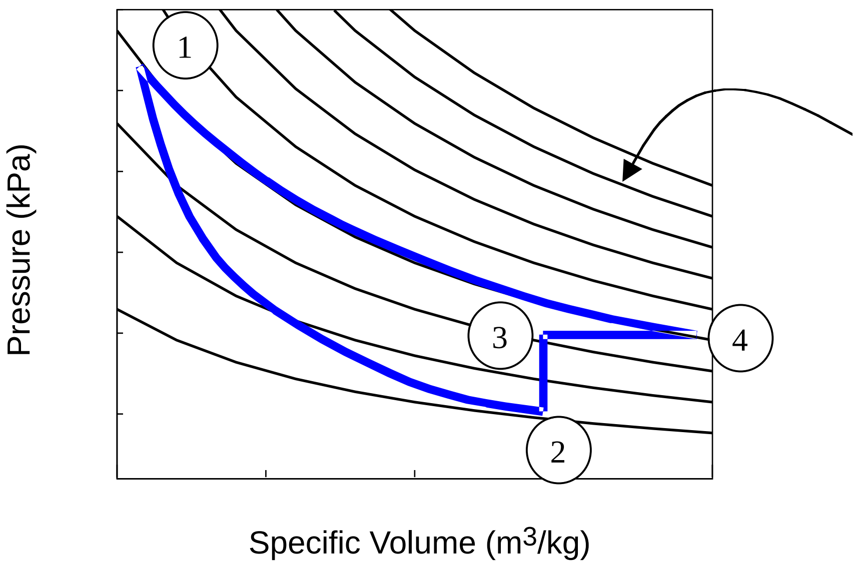
LESS WORK REQUIRED TO COMPRESS THE GAS TO THE SAME PRESSURE VIA A Q-S PROCESS.
∴ MORE EFFICIENT

T5 SOLUTIONS (Waitz)

a) Draw a thermodynamic cycle on p-v and T- v diagrams consisting of

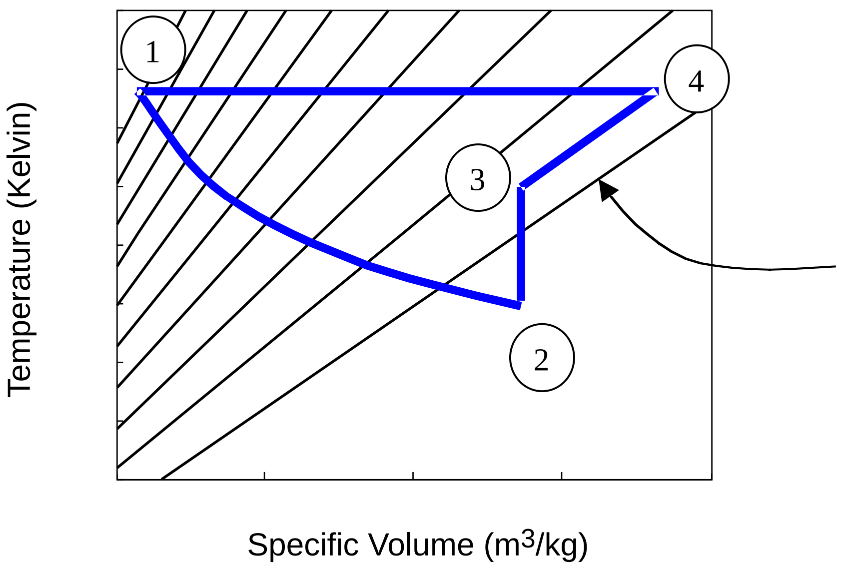
Leg 1-2: adiabatic expansion
Leg 2-3: constant volume heat addition
Leg 3-4: constant pressure expansion
Leg 4-1: isothermal compression

Assume that all processes are quasi-static and involve an ideal gas.



For 1-2, pv^{γ}=constant

adiabat is steeper than pv=constant (an isotherm)



For 1-2, $Tv^{\gamma-1}$=constant

a curved line much like the adiabat on a p-v diagram but it would be shallower

b) For each leg determine if the heat and work transfers are (+), (-), or zero.

	Q (+, -, or zero)	W (+, -, or zero)
Leg 1-2	**0**	+
Leg 2-3	+	**0**
Leg 3-4	+	+
Leg 4-1	-	-

c) Is the net work for this cycle positive or negative?

The net work for this cycle is negative The area under the expansion process is less than the area under the compression processes.

d) What common purpose might you use a cycle like this for and why?

This cycle could serve as a cooler or refrigerator. Overall it takes in energy in the form of heat from cold temperatures and expels energy in the form of heat from high temperatures. The net work for the cycle as a whole is negative, meaning that energy is put into the system to enable these transfers of heat to take place.

C2 Solutions

1. Modify the “Hello” program shown in class (Lecture C2) to display the following text on the screen:

```
Hello World
My name is Your Name
```

GNAT 3.13p (20000509) Copyright 1992-2000 Free Software Foundation, Inc.

Compiling: c:/docume~2/joeb/desktop/16070/concep~1/hello_world.adb (source file time stamp: 2003-09-10 08:55:28)

```
-----------------------------------------------------
-- Program : To Display "Hello World
--                    My Name is Joe B"
-- Programmer : Joe B
-- Date Last Modified : 09/10/2003
-----------------------------------------------------


with Ada.Text_Io;

procedure Hello_World is

begin -- Hello_World

   Ada.Text_Io.Put(Item => "Hello World ");
   Ada.Text_Io.New_Line;
   Ada.Text_Io.Put(Item => "My Name is Joe B");

end Hello_World;
```

19 lines: No errors

2. There are two errors that are seen:

i. “raised SCREEN.WIN32_FILL_SCREEN_ERROR : screen.adb:99” is seen when the output is redirected to the file.

ii. “raised SPIDER.HIT_THE_WALL: spider.adb:224” is seen when the output is only displayed on the screen

3. Write an algorithm to use the Feldman “spider package” to draw an inverted triangle as shown below. Turn in a hard copy of your code listing and an electronic copy of your code.

```
RRRRRRR
 R   R
  R R
   R
```

Problem Analysis:

The inverted triangle consists of 4 lines with the following features:

i. The top line consists of 7 symbols with no gaps between them
ii. The following lines have (n-1) blanks spaces, followed by a symbol, followed by (7-2n) blanks spaces and one symbol (if 7-2n > 0)

Algorithm:

1. Face the spider East
2. Set Spider color to Red
3. Move spider East 7 steps
4. for I in 2 .. 4 loop
    i. Set Spider color to None
    ii. Set Spider direction to South
    iii. Move Spider 1 step
    iv. Set Spider direction to West
    v. Move Spider 7 steps
    vi. Set Spider direction East
    vii. Move spider (I-1) spaces
    viii. Set Spider color to Red
    ix. Move spider one step in the same direction as last move
    x. Set Spider color to Black
    xi. Move spider (n-2*I) steps
    xii. If (n-2*I) > 0 then
        1. Set Spider color to Red
        2. Move one step
        3. Set Spider color to none
        4. move I-1 steps
5. Stop program execution

Code Listing:

GNAT 3.13p (20000509) Copyright 1992-2000 Free Software Foundation, Inc.

Compiling: c:/docume~2/joeb/desktop/adatex~1/pset1~1/spider_triangle.adb (source file time stamp: 2003-09-10 10:28:10)

```
with Spider;
procedure Spider_Triangle is
  ------------------------------------------------------------
  --| Program : This program demonstrates display an inverted triangle.
  --| Programmer : Joe B
  --| Date Last Modified : 09/10/2003
  ------------------------------------------------------------
  N : Integer;

begin -- Spider_Triangle
```

```
  -- initialise the number of lines R's per line to be 7
  N:=7;
  Spider.Start;
  -- initialize the direction to be east and set the symbol to Red
  Spider.Face(Whichway => Spider.East);
  Spider.Changecolor(Newcolor => Spider.Red);
  -- draw the top line with n Red Symbols
  for I in 1..N loop
    Spider.Step;
  end loop;
  -- the number of lines for n symbols is (n/2)+1 for n odd
  for I in 2 .. ((N/2)+1) loop
    -- move the spider down to the next line
    Spider.Face(Whichway => Spider.South);
    Spider.Changecolor(Newcolor => Spider.None);
    Spider.Step;
    -- face the opposite direction and trace back to the starting point
    Spider.Face(Whichway => Spider.West);
    for J in 1..N loop
      Spider.Step;
    end loop;
    -- turn the spider back in the right direction
    Spider.Face(Whichway => Spider.East);
    -- draw the required number of blank spaces
    for J in 1 .. I-1 loop
      Spider.Step;
    end loop;
    -- change the symbol to Red
    Spider.Changecolor(Newcolor => Spider.Red);
    Spider.Step;
    -- return the symbol to none
    Spider.Changecolor(Newcolor => Spider.None);
    -- draw the required number of blank spaces
    for J in 1 .. (N-2*I) loop
      Spider.Step;
    end loop;
    -- check to ensure that it is not the last line
    if (N-2*I > 0) then
      -- change the symbol to red and draw the symbol
      Spider.Changecolor(Newcolor => Spider.Red);
      Spider.Step;
      -- reset the symbol to none and draw the required number of blank spaces
      Spider.Changecolor(Newcolor => Spider.None);
      for J in 1 .. I-1 loop
        Spider.Step;
      end loop;
    end if;

  end loop;
  Spider.Quit;
end Spider_Triangle;
```

61 lines: No errors

C3 Solutions

1. Distance_With_Errors Listing File

GNAT 3.13p (20000509) Copyright 1992-2000 Free Software Foundation, Inc.

Compiling: c:/docume~2/joeb/desktop/adatex~1/fk3-w95/distance_with_errors.adb (source file time stamp: 1998-09-13 23:11:36)

```
WITH Ada.Text_IO;
WITH Ada.Float_Text_IO;
PROCEDURE Distance_with_Errors IS
------------------------------------------------------------
--| Finds distance, given travel time and average speed
--| Author: Michael eldman, The George Washington University
--| Last Modified: June 1998
------------------------------------------------------------
  HowLong : Natural;
  HowFast : Float;
  HowFar  : Natural;

BEGIN -- Distance_with_Errors

  -- prompt user for hours and average speed
  Ada.Text_IO.Put
    (Item => "How long will you be driving (integer) ? ");
  Ada.Float_Text_IO.Get (Item => HowLong);
                |
>>> invalid parameter list in call (use -gnatf for details)

  Ada.Text_IO.Put
    (Item => "At what speed (miles per hour, integer)?");
  Ada.Float_Text_IO.Get (Item => HowFast);

  -- compute distance driven
  HowFast := HowLong * HowFar;
                 |
>>> expected type "Standard.Float"
>>> found type "Standard.Integer"


  -- display results
  Ada.Text_IO.Put (Item => "You will travel about ");
  Ada.Float_Text_IO.Put (Item => HowFar);
                |
>>> invalid parameter list in call (use -gnatf for details)
>>> possible missing instantiation of Text_IO.Integer_IO

  Ada.Text_IO.Put (Item => " miles");
  Ada.Text_IO.New_Line;

END Distance_with_Errors;

33 lines: 5 errors
```

2. Distance_With_Errors with bug fixes

GNAT 3.13p (20000509) Copyright 1992-2000 Free Software Foundation, Inc.

Compiling: c:/docume~2/joeb/desktop/adatex~1/fk3-w95/distance_with_errors.adb (source file time stamp: 2003-09-10 10:41:38)

```
WITH Ada.Text_IO;
with Ada.Float_Text_Io;
with Ada.Integer_Text_IO;
PROCEDURE Distance_with_Errors IS
------------------------------------------------------------
--| Finds distance, given travel time and average speed
--| Author: Michael eldman, The George Washington University
--| Last Modified: June 1998
------------------------------------------------------------
  HowLong : Natural;
  HowFast : Float;
  HowFar  : Natural;

BEGIN -- Distance_with_Errors

  -- prompt user for hours and average speed
  Ada.Text_IO.Put
    (Item => "How long will you be driving (integer) ? ");
  Ada.Integer_Text_IO.Get (Item => HowLong);
  Ada.Text_IO.Put
    (Item => "At what speed (miles per hour, integer)?");
  Ada.Float_Text_IO.Get (Item => HowFast);

  -- compute distance driven
  HowFar:= HowLong * Integer(HowFast);

  -- display results
  Ada.Text_IO.Put (Item => "You will travel about ");
  Ada.Integer_Text_IO.Put (Item => HowFar);
  Ada.Text_IO.Put (Item => " miles");
  Ada.Text_IO.New_Line;

END Distance_with_Errors;

```

34 lines: No errors

3. Write an algorithm to
   a. Accept the weight of the user (in kilograms)
   b. Compute the equivalent weight in pounds
   c. Display “weight_in_kg” kg = “weight_in_pounds” lb

Where weight_in_kg is the entered value and weight_in_pounds is the computed value.

1. Prompt the user to enter his/her weight.
2. Read the user input.
3. Convert the weight from kilograms into pounds using the formula
   1 Pound = 0.453592 kilograms

   Weight in pounds = Weight in Kilograms / 0.453592

4. Display the output to the user in the weight_in_kg” kg = “weight_in_pounds” lb format.

4. Code listing of the implementation of the algorithm.

GNAT 3.13p (20000509) Copyright 1992-2000 Free Software Foundation, Inc.

Compiling: c:/docume~2/joeb/desktop/16070/concep~1/convert_weight.adb (source file time stamp: 2003-09-10 18:46:40)

```
----------------------------------------------------
-- Program : To Convert the user weight in kilograms
--           into pounds.
-- Programmer : Joe B
-- Date Last Modified : 09/10/2003
-----------------------------------------------------


with Ada.Text_Io;
with Ada.Float_Text_Io;

procedure Convert_Weight is

   Weight_In_Kg, Weight_In_Lb : Float;
   -- set the conversion factor to convert between kilograms and pounds
   Conversion_Factor : constant Float := 0.453592;

begin -- Convert_Weight
   -- get user input
   Ada.Text_Io.Put(Item => "Please Enter Your Weight in Kilograms ");
   Ada.Float_Text_Io.Get(Item => Weight_In_Kg);
   Ada.Text_Io.Skip_Line;

   Ada.Text_Io.New_Line;
   -- perform the conversion
   Weight_in_lb := Weight_in_Kg / Conversion_Factor;

   -- display the computed result to the user
   Ada.Float_Text_Io.Put(Item => Weight_In_Kg, Fore => 4, Aft => 3, Exp => 0);
   Ada.Text_Io.Put(Item => " kg = ");

   Ada.Float_Text_Io.Put(Item => Weight_In_lb, Fore => 4, Aft => 3, Exp => 0);
   Ada.Text_Io.Put(Item => " lb");

end Convert_Weight;
```

35 lines: No errors

C4 Solutions

1. Convert ‘2 + 3 = 5’ into ASCII

‘2’	-	50
‘ ’	-	32
‘+’	-	43
‘ ’	-	32
‘3’	-	51
‘=’	-	61
‘5’	-	53

2. Convert the following binary numbers into hexadecimal.

a.	<u>0000</u>	<u>1111</u>	<u>0000</u>	<u>1111</u>		
	0	F	0	F		
b.	<u>0011</u>	<u>0011</u>	<u>0000</u>	<u>0000</u>	<u>1000</u>	<u>0000</u>
	3	3	0	0	8	0
c.	<u>0000</u>	<u>1010</u>	<u>1010</u>	<u>0000</u>		
	0	A	A	0		