Planning Your Mission to Mars, the Red Planet

[Part 6]

David J. Ritchie, Tim Winder, Craig Weber

[Note]

This was independently written to work with the Mars Simulation unit available from Interact, 5937 Darwin Court, Suite 106, Carslbad, CA 92008.
Phone: (800)359-0961 Web: http://www.teachinteract.com/

The Mars Simulation is a copyrighted product of Interact.
They have not reviewed or endorsed this document.

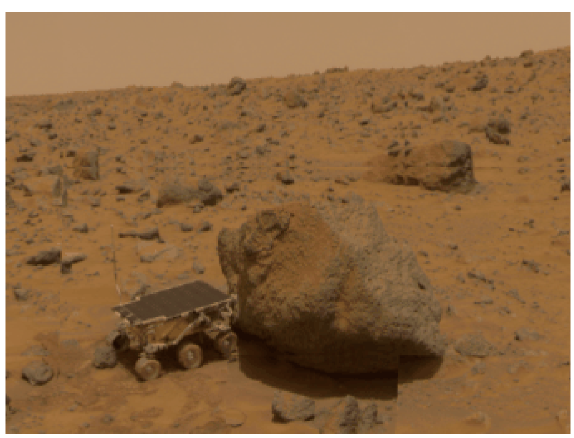
Mars Mission Science Camp
School District 203, Naperville, IL

July 17, 1998

Summary

This is the sixth session in which you will use the computer programming language "perl" to:

plan your mission to Mars, the red planet.



---

http://www.jpl.nasa.gov/files/images/browse/81958b.gif

1.0 Schedule - Where Are We in Our Mars Simulation?

The following table shows what we have accomplished (**x**) and what we are about to do (>):

Day	•	Computer Session	•	Class Session
Monday:	x	Learn Basics of Perl	x	Human Factors: Food, Water, Oxygen
Tuesday:	x	Add human factors to simulation	x	Perils of the Journey: Microgravity versus
	x	Run the simulation.	x	Artificial Gravity. Exercise. Solar Flares.
Wednesday:	x	Add perils to simulation.	x	Life on Mars. Where to land. Wind, Water,
	x	Run the simulation.	x	CO2, hard/soft, high/low,
Thursday:	x	Add landing site to simulation	x	Trajectory: Direct vs Slingshot around Venus
Friday:	>	Add trajectory to simulation		
	>	Run the simulation.		

Yesterday, you considered how to get to Mars. I will call one the "Slingshot" (around Venus) approach and the other the "Direct" approach. I have made a table to compare the two approaches.

Aspect	Direct Approach	Slingshot Approach
Gravity	No gravity	Artificial Gravity
Mars-bound	8 months	11 months
Earth-bound	5 months	8 months
Time on Mars	20-30 days	40 days
Organization	Crew and Cargo	Two on a tether
Miscellanous	Separate Landers Constructed on space Station: 15 months	Lander not specified Construction not specified

I have put a trajectory selection option into the simulation and extended the Simulation Story and Dictionary. I have ignored the 15 month construction time in the case of the Direct Approach as being irrelevant.

2.0 Evaluation Sheet

This is the first time that we have done a computer simulation as part of the Advanced Mars Camp. Please tell us how to improve. Keep in mind that we have a limited budget Suggest practical things. We'd all love to go to JPL for a week but that's unlikely.

2.1 Things that were good (keep these)

2.2 Things that were not good (drop or revise--how?)

2.3 Any other suggestions for improving the sesssions

3.0 The Simulation Story

We need a simulation story to guide us. The story describes the facts we are going to take into account in our simulation and how we are going to do it. So far: Human Factors, MicroGravity Disabilities, and now Landing Site Selection

3.1 The Human Factors and Perils Chapter

The mission lasts for some months. We will treat months as having 30 days each. A mission has a "sponsoring country" whose name can be specified.

There are four to six crew members with the numbers of men and women to be specified as input into the simulation. There can be one or two scientists and one or two military officers. The second scientist or military officer is on the mission provided he or she has been given a name.

The daily food, drinking water, and oxygen requirements for each woman crew member are 2.55 lbs, 4.25 lbs, and 1.70 lbs, respectively. The daily food, drinking water, and oxygen requirements for each man crew member are 3 lbs, 5 lbs, and 2 lbs, respectively.

We begin with a total amount of food, drinking water, and oxygen. Each day we reduce the total amount of food, drinking water, and oxygen. If we are recycling everything, there is nothing more to say.

Each person now has a health factor. The health factor starts at 100.

Each person also has a daily exercise counter. If this is set to 1, the person exercises every day. If set to 2, every second day, etc. Each day the person exercises 0.1 is **added** to their health factor. Each day the person exercises the daily food, water, and oxygen usage is multiplied by 1.05–increasing the daily food, water and oxygen consumption per exercising person by 5%.

The mission now has a gravity indicator. This indicator is "true" if the ships containing the human beings have artificial gravity. Otherwise, it is false meaning that there is **no** gravity. If there is no gravity, then 0.075 is **subtracted** from the person's health factor each day so that exercise wins a bit over the lack of gravity.

3.2 The Site Selection Chapter

The mission team can indicate its site preferences in four different categories: wind, firmness, terrain, and composition.

Category	Pref A	Pref B	Pref C	Pref D
Wind	Low	Med	High	
Firmness	Soft	Med	Hard	
Terrain	Canyons	Plains	Mountains	
Composition	Water	CO_2	Water & CO_2	Neither

Still, a certain randomness may occur and the mission team should expect that they may not get a site with the characteristics they want. The consequences which may occur with a certain probability are as follows:

1. High winds on a soft mountain site have a 50% probability that the lander will tip over so that extensive repairs are required. This delays your departure, resulting in your missing your launch window and having to stay on Mars five months. Mission payback is decreased by 50%.
2. Water and / or CO_2 in a soft canyon site may result in the discovery of primitive life forms. The first day this occurs, the mission payback is doubled. After that first time, each discovery increases the mission payback by 10%.
3. Low winds on soft or medium plains almost always results in a successful landing with no ill effects. Discovery probability is low, however.
4. Hard mountain sites are generally not good though discovery probability may be higher because of possible fossils.
5. Medium wind on areas having neither water nor CO_2 are alright as successful landing sites but discovery probability is low.

The team can choose one of the specific spots with certainty. The specific spot will hopefully have the preferred characteristics requested by the team.

If the Direct Approach Trajectory is chosen, the landing period will commence 8 months (240 days) after the start of the mission. If the Slingshot Trajectory Approach is chosen, it will occur 11 months (330 days) after the start of the mission.

If the Direct Approach is chosen, the time on Mars can be adjusted to be between 20 and 30 days. If the Slingshot Approach is chosen, the time on Mars is 40 days. Each day there is a random chance of discovery. Each day a fossil is discovered, the mission payback is increased by 7%. If primitive life forms are discovered, the mission payback is doubled once.

Gravity is switched on while the team is on Mars. Afterwards, it is put back to whatever it was before the landing.

I assume that the food, oxygen, and water are distributed between the two separate ships on the voyage. In the Direct Approach, there is no access to these resources until the Mars-bound voyage is completed. In the Slingshot Approach, there is access to all the resources.

At the beginning of the mission in the Direct Approach, the total food, water, and oxygen are divided into amounts that are sized in the same proportion as the lengths of the Mars-bound and Earth-bound trips. That is, in the Direct Approach, the food is split into 8/(5+8) and 5/(5+8) chunks. The 8/(5+8) chunk is put aboard the crew ship. The 5/(5+8) chunk is put aboard the cargo ship. In the Slingshot approach, the resources are split 50-50 between the two ships.

There is a 0.1% (one-tenth of one percent) probability that one of the ships is lost during the voyage. Whether or not that happens is computed on the day of the landing on Mars. If it happens, then the food, oxygen, and water are reduced to what is in the crew ship at that point. The mission must return to Earth on those remaining resources. The payback is reduced by 30%.

4.0 The Simulation Dictionary

This is a list of the variables in the simulation. It is somewhat out of date. See the code.

4.1 Mission Parameters

`$MisMonths` .... The length of the mission in months.

`$MisDays` .... The length of the mission in days.

`$Day` .... The particular day of the mission.

`$MisCountry` .... The name of the country sponsoring the mission

`$MisGravity` .... The mission gravity indicator. If true, the mission has artificial gravity. Otherwise, it has **no** gravity.

`$MisLandDay` .... The day of the landing

`$MisMarsDur` .... The number of days of the stay on Mars. This is a number between 1 and 40.

`$MisPayback` .... The payback for a successful mission. It is initially set at 1000 and is increased or decreased according to circumstances.

4.2 People

`$SciName` .... The name of the Scientist

`$SciMW` .... ="M" if scientist is a man; = "W" if scientist is a woman

`$SciH` .... The health of the scientist begining at 100 and increasing or decreasing as situations occur.

`$SciEF` .... The exercise frequency of the scientist: 1 => the person exercises every day, 2 => every other day, etc.

`$Sci2Name` .... The name of the second scientist. If name is "", then the scientist is not on the mission.

`$Sci2MW` .... ="M" if second scientist is a man; = "W" if second scientist is a woman

`$Sci2H` .... The health of the second scientist begining at 100 and increasing or decreasing as situations occur.

`$Sci2EF` .... The exercise frequency of the second scientist: 1 => the person exercises every day, 2 => every other day, etc.

`$MedName` .... The name of the Medical Officer

`$MedMW` .... ="M" if Medical Officer is a man; = "W" if Medical Officer is a woman

`$MedH` .... The health of the scientist begining at 100 and

	increasing or decreasing as situations occur.
`$MedEF`	The exercise frequency of the Medical Officer: 1 => the person exercises every day, 2 => every other day, etc.
`$MilName`	The name of the Military Officer
`$MilMW`	="M" if Military Officer is a man; = "W" if Military Officer is a woman
`$MilH`	The health of the Military Officer begining at 100 and increasing or decreasing as situations occur.
`$MilEF`	The exercise frequency of the Military Officer: 1 => the person exercises every day, 2 => every other day, etc.
`$Mil2Name`	The name of the second Military Officer
`$Mil2MW`	="M" if second Military Officer is a man; = "W" if second Military Officer is a woman
`$Mil2H`	The health of the second Military Officer begining at 100 and increasing or decreasing as situations occur.
`$Mil2EF`	The exercise frequency of the second Military Officer: 1 => the person exercises every day, 2 => every other day, etc.
`$MisName`	The name of the Mission Commander
`$MisMW`	="M" if Mission Commander is a man; = "W" if Mission Commander is a woman
`$MisH`	The health of the scientist begining at 100 and increasing or decreasing as situations occur.
`$MisEF`	The exercise frequency of the Mission Commander: 1 => the person exercises every day, 2 => every other day, etc.

4.3 Human Factors

`$TotalFood`	The total food on the mission
`$TotalWater`	The total water on the mission
`$TotalOxygen`	The total oxygen on the mission
`$DailyFoodW`	The daily food consumed by the average woman
`$DailyFoodM`	The daily food consumed by the average man
`$DailyWaterW`	The daily water consumed by the average woman
`$DailyWaterM`	The daily water consumed by the average man

`$DailyOxygenW` .... The daily oxygen consumed by the average woman

`$DailyOxygenM` .... The daily oxygen consumed by the average man

4.4 Site Factors

`$Site` .... Variable which holds the specific site chosen:

0 = Anywhere

1 = Rim of Polar Ice Cap

2 = Volcanic Region

3 = Area: Long 250° Lat 55° N.

4 = Area: Long 103° Lat 8° N - Tharsis region

5 = Area: Long 345° Lat 50° N - Cydonia region

6 = Volcano Olympus Mons

7 = Mariner Valley

`$Wind` .... Wind preference: 0, 1, or 2 for low, med., or high

`$Firm` .... Firmness preference: 0, 1, or 2 for soft, med., or hard

`$Terr` .... Terrain preference: 0, 1, or 2 for Canyons, Plains, or Mountains

`$Comp` .... Composition preference: 0, 1, 2, 3 for neither, water, CO_2, or both.

5.0 Human Factors, Perils, Site, & Trajectory Simulation

```
#    Example p6s50e1.pl
#
#    Mission Parameters
     $MisCountry = "Brazil";
     $MisPayback = 1000;
     $MisTraj    = 0;
     $MisMarsDur = 25;
#
#    Resources - must increase these -- just examples
     $TotalFood    = 4000.0;
     $TotalWater   = 4000.0;
     $TotalOxygen  = 4000.0;

     if ($MisTraj == 0) {
#
#        Direct Trajectory
         $MisMbMonths = 8;
         $MisEbMonths = 5;
         $MisGravity = 0;
         if ($MisMarsDur < 20) {$MisMarsDur = 20};
         if ($MisMarsDur > 30) {$MisMarsDur = 30};
     } elsif ($MisTraj == 1) {
#
#        Slingshot Trajectory
         $MisMbMonths = 11;
         $MisEbMonths = 8;
         $MisGravity = 1;
         $MisMarsDur  = 40;
     } else {
         print ("Improper Trajectory Value $MisTraj");
     }   # end if ($MisTraj)
```

```
#
#    Site Parameters and Preferences
     $Site = 0;
     $Wind = 2;
     $Firm = 2;
     $Terr = 2;
     $Comp = 0;
#
#    People
     $SciName  = "Mary Smith";
     $SciMW    =  "W";
     $SciH     = 100;
     $SciEF    =   1;
     $Sci2Name = "Jeff Slattery";
     $Sci2MW   = "M";
     $Sci2H    = 100;
     $Sci2EF   =   1;
     $MedName  = "James Roberts";
     $MedMW    = "M";
     $MedH     = 100;
     $MedEF    =   1;
     $MilName  = "Susan Limon";
     $MilMW    = "W";
     $MilH     = 100;
     $MilEF    =   1;
     $Mil2Name = "Reggie DeBates";
     $Mil2MW   = "M";
     $Mil2H    = 100;
     $Mil2EF   =   5;
     $MisName  = "Kenneth Bates";
     $MisMW    = "M";
     $MisH     = 100;
     $MisEF    =   1;
```

```
#
#    say how much a woman crew member needs each day
     $DailyFoodW   =   2.55;
     $DailyWaterW  =   4.25;
     $DailyOxygenW =   1.70;
#
#    say how much a man crew member needs each day
     $DailyFoodM   =   3.00;
     $DailyWaterM  =   5.00;
     $DailyOxygenM =   2.00;
#
#    an array with the name of the possible sites
     @SiteNames = ('Anywhere',
                   'Rim of Polar Ice Cap',
                   'Volcanic Region',
                   'Area: Long 250° Lat 55° N.',
                   'Area: Long 103° Lat  8° N - Tharsis',
                   'Area: Long 345° Lat  50° N - Cydonia',
                   'Volcano Olympus Mons',
                   'Mariner Valley');
     @WindNames = ('low','medium','high');
     @FirmNames = ('soft','medium','hard');
     @TerrNames = ('Canyons','Plains','Mountains');
     @CompNames = ('Sand','Water','Frozen CO2',
                   'Water and Frozen CO2');
#    Discovery Probabilities Initial
     $MisFosDiscProb = 0.01;
     $MisAnyDiscProb = 0.01;
     $MisLifeDiscProb = 0.01;
     $MisDiscLife = 0;
#    some places to hold current exercise counts
     $SciEFc  = $SciEF;
     $Sci2EFc = $Sci2EF;
     $MedEFc  = $MedEF;
     $MilEFc  = $MilEF;
     $Mil2EFc = $Mil2EF;
     $MisEFc  = $MisEF;
```

```
#
#    compute things about mission
     $MisMbDays = $MisMbMonths * 30;
     $MisEbDays = $MisEbMonths * 30;
#
#    distribute the food, water, oxy between ships
#    compute total available quantities at start of mission
     if ($MisTraj == 0 ) {
#
#        Direct Trajectory
         $Ship1Food = $MisMbMonths/($MisMbMonths +
                      $MisEbMonths) * $TotalFood;
         $Ship2Food = $TotalFood - $Ship1Food;
         $Ship1Water = $MisMbMonths/($MisMbMonths +
                       $MisEbMonths) * $TotalWater;
         $Ship2Water = $TotalWater - $Ship1Water;
         $Ship1Oxygen = $MisMbMonths/($MisMbMonths +
                        $MisEbMonths) * $TotalOxygen;
         $Ship2Oxygen = $TotalOxygen - $Ship1Oxygen;
         $TotalAvailFood = $Ship1Food;
         $TotalAvailWater = $Ship1Water;
         $TotalAvailOxygen = $Ship1Oxygen;
     } elsif ($MisTraj == 1 ) {
         $Ship1Food = 0.5 * $TotalFood;
         $Ship2Food = $TotalFood - $Ship1Food;
         $Ship1Water = 0.5 * $TotalFood;
         $Ship2Water = $TotalWater - $Ship1Water;
         $Ship1Oxygen = 0.5 * $TotalOxygen;
         $Ship2Oxygen = $TotalOxygen - $Ship1Oxygen;
         $TotalAvailFood = $Ship1Food + $Ship2Food;
         $TotalAvailWater = $Ship1Water + $Ship2Water;
         $TotalAvailOxygen = $Ship1Oxygen + $Ship2Oxygen;
     } else {
         print ("Improper Trajectory Value $MisTraj");
     }   # end if ($MisTraj)
```

```
#---------------------------------------------------
#    Beginning the Simulation
     print ("Beginning Simulation\n");
     print ("----Mission for $MisCountry----\n");
     print ("Mission Commander $MisName\n");
     print ("Scientist         $SciName\n");
     print ("Scientist         $Sci2Name\n") if ($Sci2Name);
     print ("Military Officer  $MilName\n");
     print ("Military Officer  $Mil2Name\n") if ($Mil2Name);
     print ("Medical Officer   $MedName\n");
#----------------------------------------------------
#    for each day of the Mars bound mission
#    exercise, deal with gravity
#    eat, drink, breathe and do Mars things
     foreach $SegDay (1..$MisMbDays) {
         $Day = $SegDay;
         Exercise();
         HandleGravity();
         EatFood();
         DrinkWater();
         BreatheOxygen();
#
#        print the day every 30 days to show we're working
         if ($Day%30 == 0) { print ("Day: $Day\n"); }
     }   # end foreach $Day
#
#    Ship Problem - see if the ship survives
     if (rand(100.0) > 0.1) {
#
#    ship survived - if the Direct Traj, move the cargo
#    nothing to do in Slingshot Trajectory
     if ($MisTraj == 0) {
         $TotalAvailFood += $Ship2Food;
         $TotalAvailWater += $Ship2Water;
         $TotalAvailOxygen += $Ship2Oxygen;
     }
```

```
#
#    Do Mars Landing or Flyby
        $MarsDay = 1;
        $MisAnyDiscProb = 0.1;
        $Day = $MisMbDays + $MarsDay;
        $MisGravitySave = $MisGravity;
        $MisGravity = 1;
        PickSite();
        print ("Day: $Day. Landing on Mars...\n");
        print ("          Site is $SiteNames[$Site]\n");
        print ("Wind is $WindNames[$RealWind]\n",
               "Firmness is $FirmNames[$RealFirm]\n",
               "Terrain is $TerrNames[$RealTerr]\n",
               "Composition is $CompNames[$RealComp]\n");
#
#       Do consequences of landing
#       Low Winds Soft or Medium Plain
        if ($RealWind == 0 &&
            $RealFirm < 2 && $RealTerr ==1)
            {$MisAnyDiscProb = 0.02};
#       Hard Mountains
        if ($RealTerr == 2 && $RealFirm == 2)
           {$MisFosDiscProb = 0.3};
#       Medium or High Wind and neither CO2 nor Water
        if ($RealWind >= 1 && $RealComp == 0)
           {$MisAnyDiscProb = 0.01};
#       High Winds on a Soft Mountain
        if ($RealWind == 2 && $RealFirm == 0 &&
            $RealTerr == 2 )
        {
            if (rand(1.0) > 0.5) {
                print ("Day: $Day. Your lander tipped.\n",
                   "          Plan on another 5 months!\n");
                $MisMarsDur += 5*30;
                $MisPayback = $MisPayback * 0.5;
            }   # end if (rand(1.0) > 0.5)
        }   # end if ($RealWind, etc.)
#       Soft Canyon with CO2
        if ($RealFirm == 0 && $RealTerr == 0
```

```
            && $RealComp == 3)  {
            $MisLifeDiscProb = 0.4;
            print ("You're in a soft canyon site",
                   " with water and Frozen CO2!",
                   " Discovery chances are good!\n");
        }   #   end if ($RealFirm...
    } else {
        #   else if the ship did not survive
        $MisAnyDiscProb = 0.01;
        $MisPayback = $MisPayback * 0.3;
        print ("Your second ship was lost.  ",
               "People are safe but the supplies",
               "are gone.  Your mission commander, \n",
               "$MisName, has decided to scrub the mission",
               "and go back to Earth on the supplies you",
               "have left. Unfortunately, Payback\n",
               "is now $MisPayback. Good Luck!\n");
#
#       if it is the slingshot approach, we lose half
#       our supplies - direct approach simply does nothing
#       as there is nothing to transfer.
        if ($MisTraj == 1) {
            $TotalAvailFood = $TotalAvailFood/2.0;
            $TotalAvailWater = $TotalAvailWater/2.0;
            $TotalAvailOxygen = $TotalAvailOxygen/2.0;
        }
    }
#
#    for each day we are on Mars
    foreach $SegDay (1..$MisMarsDur) {
    $Day = $MisMbDays + $SegDay;
    DoMarsDay();
    }
#
#    do take off day
    $SegDay = $MisMarsDur;
    $Day = $MisMbDays + $SegDay;
    print ("Day: $Day. Taking off from Mars...\n");
#
```

```
#    restore gravity flag
     $MisGravity = $MisGravitySave;
#
#    for each day of the Earth bound mission
#    exercise, deal with gravity
#    eat, drink, breathe and do Mars things
     foreach $SegDay (1..$MisEbDays) {
         $Day = $MisMbDays + $MisMarsDur + $SegDay;
         Exercise();
         HandleGravity();
         EatFood();
         DrinkWater();
         BreatheOxygen();
#
#        print the day every 30 days to show we're working
         if ($Day%30 == 0) { print ("Day: $Day\n"); }
     }   # end foreach $Day
$MisDays = $Day;
```

```
#
#    print status at the end of mission
#    resources left
     printf ("---------Status At End of Mission---------\n");
     printf ("       Time: %18.1f (days)\n",$MisDays);
     printf ("  Food Left: %18.1f\n Water Left: %18.1f\n  Oxy
     Left: %18.1f\n",
           $TotalAvailFood, $TotalAvailWater,
     $TotalAvailOxygen);
     printf ("    Payback: %18.1f \n",$MisPayback);
#
#    print status of people
     printf ("Scientist:  %18.18s Health: %18.1f\n",
                   $SciName, $SciH);
     printf ("Scientist:  %18.18s Health: %18.1f\n",
                   $Sci2Name,$Sci2H) if ($Sci2Name);
     printf ("Med. Off.:  %18.18s Health: %18.1f\n",
                   $MedName,$MedH);
     printf ("Mil. Off.:  %18.18s Health: %18.1f\n",
                   $MilName,$MilH);
     printf ("Mil. Off.:  %18.18s Health: %18.1f\n",
                   $Mil2Name,$Mil2H) if ($Mil2Name);
     printf ("Mis. Cmd.:  %18.18s Health: %18.1f\n",
                   $MisName,$MisH);
     exit;
```

```
############ Beginning of all our subroutines##########
sub  PickSite {
     if ($Site == 0) {
         $Site = rand(6.0) + 1.0;
     }
#    get four random numbers to help pick
     $PickWind = rand(10.0);
     $PickFirm = rand(10.0);
     $PickTerr = rand(10.0);
     $PickComp = rand(10.0);
#
#    Now pick team's preference 70 percent of the time
     if ($PickWind > 3.0 ) {
         $RealWind = $Wind ;
     } else {
         $RealWind = rand(3)%3;
     }
     if ($PickFirm > 3.0 ) {
         $RealFirm = $Firm ;
     } else {
         $RealFirm = rand(3)%3;
     }
     if ($PickTerr > 3.0 ) {
         $RealTerr = $Terr ;
     } else {
         $RealTerr = rand(3)%3;
     }
     if ($PickComp > 3.0 ) {
         $RealComp = $Comp ;
     } else {
         $RealComp = rand(4)%4;
     }
     return;
}
```

```
sub  DoMarsDay {
#
#    Increase the probability of discovery every day
#    as we are learning about the place
     $MisAnyDiscProb += 0.05;
#
#    Life
     if (rand() > $MisLifeDiscProb) {
         if ($MisDiscLife == 0) {
             print ("The $MisCountry mission has",
                    "found a primitive form of life!!!!\n");
             $MisPayback = $MisPayback * 2;
             $MisDiscLife = 1;
         } else {
             print ("Additional life species discovered.\n");
             $MisPayback += $MisPayback * 0.1;
         }
     }
#    Fossils
     if (rand() > $MisFosDiscProb) {
         print ("The $MisCountry mission has",
                "found some unusual fossils.\n");
         $MisPayback += $MisPayback * 0.07;
     }
#    Any
     if (rand() > $MisAnyDiscProb) {
         print ("The $MisCountry mission has",
                "made some puzzling discoveries.\n");
         $MisPayback += $MisPayback * 0.03;
     }
     return;
}
```

```
sub  HandleGravity {
#
#    if no gravity then adjust health down
     if ($MisGravity == 0) {
         $SciH  = $SciH  - 0.075;
         $Sci2H = $Sci2H - 0.075 if ($Sci2Name ne "");
         $MedH  = $MedH  - 0.075;
         $MilH  = $MilH  - 0.075;
         $Mil2H = $Mil2H - 0.075 if ($Mil2Name ne "");
         $MisH  = $MisH  - 0.075;
     }
     return;
}
```

```
sub  Exercise {
#
#    Scientist: Count down exercise frequency
#    when get to 0, add 0.1 to health and reset exer freq cnt
     $SciEFc = $SciEFc - 1;
     $SciH = $SciH + 0.1 if ($SciEFc == 0);
     $SciEFc = $SciEF if ($SciEFc == 0);
#    Second Scientist
     if ($Sci2Name ne "") {
         $Sci2EFc = $Sci2EFc - 1;
         $Sci2H = $Sci2H + 0.1 if ($Sci2EFc == 0);
         $Sci2EFc = $Sci2EF if ($Sci2EFc == 0);
     }
#    Medical Officer
     $MedEFc = $MedEFc - 1;
     $MedH = $MedH + 0.1 if ($MedEFc == 0);
     $MedEFc = $MedEF if ($MedEFc == 0);
#    Military Officer
     $MilEFc = $MilEFc - 1;
     $MilH = $MilH + 0.1 if ($MilEFc == 0);
     $MilEFc = $MilEF if ($MilEFc == 0);
#    Second Military Officer
     if ($Mil2Name ne "") {
         $Mil2EFc = $Mil2EFc - 1;
         $Mil2H = $Mil2H + 0.1 if ($Mil2EFc == 0);
         $Mil2EFc = $Mil2EF if ($Mil2EFc == 0);
     }
#    Mission Commander
     $MisEFc = $MisEFc - 1;
     $MisH = $MisH + 0.1 if ($MisEFc == 0);
     $MisEFc = $MisEF if ($MisEFc == 0);
     return;
}
```

```
sub  EatFood {
#
#    Initialize Daily Food Usage
     $DailyFood = 0.0;
#
#    Assume Scientist is just regularly hungry
#    and then see if Scientist just finished exercising
#    and increase hunger accordingly
     $Hungry = 1.00;
     if ($SciEF eq $SciEFc) {$Hungry = 1.05;}
     if ($SciMW eq "M") { $DailyFood += $DailyFoodM*$Hungry;}
     else               { $DailyFood += $DailyFoodW*$Hungry;}
     if (!$Sci2Name) {
     $Hungry = 1.00;
     if ($Sci2EF eq $Sci2EFc) {$Hungry = 1.05;}
         if ($Sci2MW eq "M") {$DailyFood +=
                              $DailyFoodM*$Hungry;}
         else                {$DailyFood +=
                              $DailyFoodW*$Hungry;}
     }
     $Hungry = 1.00;
     if ($MedEF eq $MedEFc) {$Hungry = 1.05;}
     if ($MedMW eq "M") {$DailyFood += $DailyFoodM*$Hungry;}
     else               {$DailyFood += $DailyFoodW*$Hungry;}
     $Hungry = 1.00;
     if ($MilEF eq $MilEFc) {$Hungry = 1.05;}
     if ($MilMW eq "M") {$DailyFood += $DailyFoodM*$Hungry;}
     else               {$DailyFood += $DailyFoodW*$Hungry;}
     if (!$Mil2Name) {
         $Hungry = 1.00;
         if ($Mil2EF eq $Mil2EFc) {$Hungry = 1.05;}
         if ($Mil2MW eq "M") {$DailyFood +=
                              $DailyFoodM*$Hungry;}
         else                {$DailyFood +=
                              $DailyFoodW*$Hungry;}
     }
     $Hungry = 1.00;
     if ($MisEF eq $MisEFc) {$Hungry = 1.05;}
     if ($MisMW eq "M") {$DailyFood += $DailyFoodM*$Hungry;}
```

```
    else                {$DailyFood += $DailyFoodW*$Hungry;}
    $TotalAvailFood = $TotalAvailFood - $DailyFood;
    if ($TotalAvailFood <= 0 ) {
        print ("Day $Day:  No food available!\a\n",
             "          TF: $TotalFood TAF
    $TotalAvailFood\n");
        exit;
    }
    return;
}
```

```
#    Subroutine DrinkWater
sub  DrinkWater {
#
#    Get total daily water according to man or woman
     $DailyWater = 0.0;
     $Thirsty = 1.00;
     if ($SciEF eq $SciEFc) {$Thirsty = 1.05;}
     if ($SciMW eq "M") { $DailyWater +=
                          $DailyWaterM*$Thirsty;}
     else               { $DailyWater +=
                          $DailyWaterW*$Thirsty;}
     if (!$Sci2Name) {
         $Thirsty = 1.00;
         if ($Sci2EF eq $Sci2EFc) {$Thirsty = 1.05;}
         if ($Sci2MW eq "M") {$DailyWater +=
                              $DailyWaterM*$Thirsty;}
         else                {$DailyWater +=
                              $DailyWaterW*$Thirsty;}
     }
     $Thirsty = 1.00;
     if ($MedEF eq $MedEFc) {$Thirsty = 1.05;}
     if ($MedMW eq "M") {$DailyWater +=
                         $DailyWaterM*$Thirsty;}
     else               {$DailyWater +=
                         $DailyWaterW*$Thirsty;}
     $Thirsty = 1.00;
     if ($MilEF eq $MilEFc) {$Thirsty = 1.05;}
     if ($MilMW eq "M") {$DailyWater +=
                         $DailyWaterM*$Thirsty;}
     else               {$DailyWater +=
                         $DailyWaterW*$Thirsty;}
     if (!$Mil2Name) {
         $Thirsty = 1.00;
         if ($Mil2EF eq $Mil2EFc) {$Thirsty = 1.05;}
         if ($Mil2MW eq "M") {$DailyWater +=
                              $DailyWaterM*$Thirsty;}
         else                {$DailyWater +=
                              $DailyWaterW*$Thirsty;}
     }
```

```
     $Thirsty = 1.00;
     if ($MisEF eq $MisEFc) {$Thirsty = 1.05;}
     if ($MisMW eq "M") {$DailyWater +=
                         $DailyWaterM*$Thirsty;}
     else               {$DailyWater +=
                         $DailyWaterW*$Thirsty;}
     $TotalAvailWater = $TotalAvailWater - $DailyWater;
#
#    if we have drunk all the water, print out day and exit
     if ($TotalAvailWater <= 0 ) {
         print ("Day $Day:  No water available!\a\n",
              "          TW: $TotalFood TAW
     $TotalAvailFood\n");
         exit;
     }
     return;
}
```

```
#    Subroutine BreatheOxygen
sub  BreatheOxygen {
#
#    Get total daily oxygen according to man or woman
     $DailyOxygen = 0.0;
     $Huff = 1.00;
     if ($SciEF eq $SciEFc) {$Huff = 1.05;}
     if ($SciMW eq "M") { $DailyOxygen +=
                          $DailyOxygenM*$Huff;}
     else               { $DailyOxygen +=
                          $DailyOxygenW*$Huff;}
     if (!$Sci2Name) {
         $Huff = 1.00;
         if ($Sci2EF eq $Sci2EFc) {$Huff = 1.05;}
         if ($Sci2MW eq "M") {$DailyOxygen +=
                              $DailyOxygenM*$Huff;}
         else                {$DailyOxygen +=
                              $DailyOxygenW*$Huff;}
     }
     $Huff = 1.00;
     if ($MedEF eq $MedEFc) {$Huff = 1.05;}
     if ($MedMW eq "M") {$DailyOxygen +=
                         $DailyOxygenM*$Huff;}
     else               {$DailyOxygen +=
                         $DailyOxygenW*$Huff;}
     $Huff = 1.00;
     if ($MilEF eq $MilEFc) {$Huff = 1.05;}
     if ($MilMW eq "M") {$DailyOxygen +=
                         $DailyOxygenM*$Huff;}
     else               {$DailyOxygen +=
                         $DailyOxygenW*$Huff;}
     if (!$Mil2Name) {
         $Huff = 1.00;
         if ($Mil2EF eq $Mil2EFc) {$Huff = 1.05;}
         if ($Mil2MW eq "M") {$DailyOxygen +=
                              $DailyOxygenM*$Huff;}
         else                {$DailyOxygen +=
                              $DailyOxygenW*$Huff;}
     }
```

```
        $Huff = 1.00;
        if ($MisEF eq $MisEFc) {$Huff = 1.05;}
        if ($MisMW eq "M") {$DailyOxygen +=
                            $DailyOxygenM*$Huff;}
        else               {$DailyOxygen +=
                            $DailyOxygenW*$Huff;}
        $TotalAvailOxygen = $TotalAvailOxygen - $DailyOxygen;
#
#       if we have breathed the oxygen, print out day and exit
        if ($TotalAvailOxygen <= 0 ) {
            print ("Day $Day:  No oxygen available!\a\n",
                 "          TO: $TotalOxygen TAO
        $TotalAvailOxygen\n");
            exit;
        }
        return;
}
```