

# 70年代のマイコンホビーストに大人気だった 「STARTREK」ゲームの紹介

プログラムがスタートするとエンタープライズ号がスクロールしながら表示され、停止後数秒待つてから、命令が表示されます。

下の画面では、

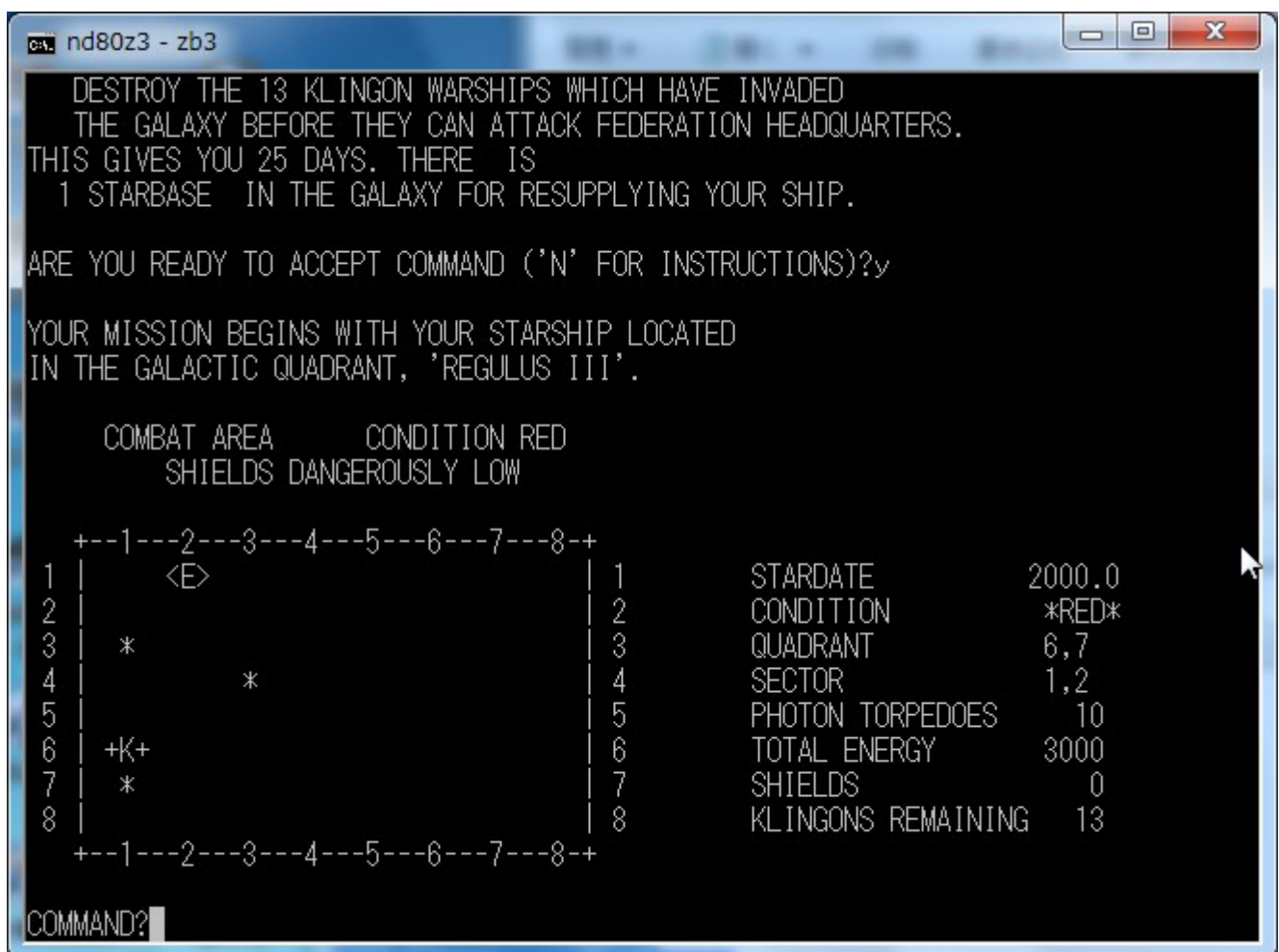
25日間で敵戦艦（KLINGON WARSHIP）13艦を破壊せよという指令が与えられています。

ARE YOU READY ACCEPT COMMAND?

にはN以外の文字を入力して [Enter] を押します。

以下、何かを入力するときは最後に [Enter] を押します。

すると下のように表示されます。



```
nd80z3 - zb3
DESTRUCT THE 13 KLINGON WARSHIPS WHICH HAVE INVADED
THE GALAXY BEFORE THEY CAN ATTACK FEDERATION HEADQUARTERS.
THIS GIVES YOU 25 DAYS. THERE IS
1 STARBASE IN THE GALAXY FOR RESUPPLYING YOUR SHIP.

ARE YOU READY TO ACCEPT COMMAND ('N' FOR INSTRUCTIONS)?y

YOUR MISSION BEGINS WITH YOUR STARSHIP LOCATED
IN THE GALACTIC QUADRANT, 'REGULUS III'.

      COMBAT AREA      CONDITION RED
      SHIELDS DANGEROUSLY LOW

+--1--2--3--4--5--6--7--8--+
1 |          <E>          | 1  STARDATE           2000.0
2 |          |          | 2  CONDITION           *RED*
3 |   *          |          | 3  QUADRANT            6,7
4 |          |          *  | 4  SECTOR              1,2
5 |          |          | 5  PHOTON TORPEDOES       10
6 |  +K+          |          | 6  TOTAL ENERGY       3000
7 |   *          |          | 7  SHIELDS              0
8 |          |          | 8  KLINGONS REMAINING    13
+--1--2--3--4--5--6--7--8--+

COMMAND? |
```

STARTREKゲームの舞台となります全宇宙は8×8の小宇宙で構成されています。  
最初に画面に表示されるのが、現在エンタープライズ号がいる小宇宙（QUADRANT）です。

画面右側の

QUADRANT 6, 7

がその小宇宙の位置を示しています。

	1	2	3	4	5	6	7	8
1								
2								
3								
4								
5								
6							(6, 7)	
7								
8								

表示されている、現在エンタープライズ号がいる小宇宙（QUADRANT）も8×8マスの座標に区切られています。

+--1--2--3--4--5--6--7--8--+									
1		<E>							1
2									2
3	*								3
4			*						4
5									5
6	+K+								6
7	*								7
8									8
+--1--2--3--4--5--6--7--8--+									

STARDATE	2000.0
CONDITION	*RED*
QUADRANT	6,7
SECTOR	1,2
PHOTON TORPEDOES	10
TOTAL ENERGY	3000
SHIELDS	0
KLINGONS REMAINING	13

<E>がエンタープライズ号です。

その位置は右のSECTOR 1, 2で示されています。

数学の座標などでは普通は(X, Y)のようにX(ヨコ)が先でY(タテ)があとというように表記しますが、STARTREKではそれとは逆で、タテが先、ヨコがあとに表示されます。

敵(KLINGON)は+K+で表示されます。

たまたま最初の場面で敵に遭遇しましたが、敵がいないこともあります。

今回の設定では64の小宇宙に敵は13ですから、敵がいない場所の方が多いことになります。

その敵を別の小宇宙にワープしながら探します。

この図では敵は(6, 1)にいます。

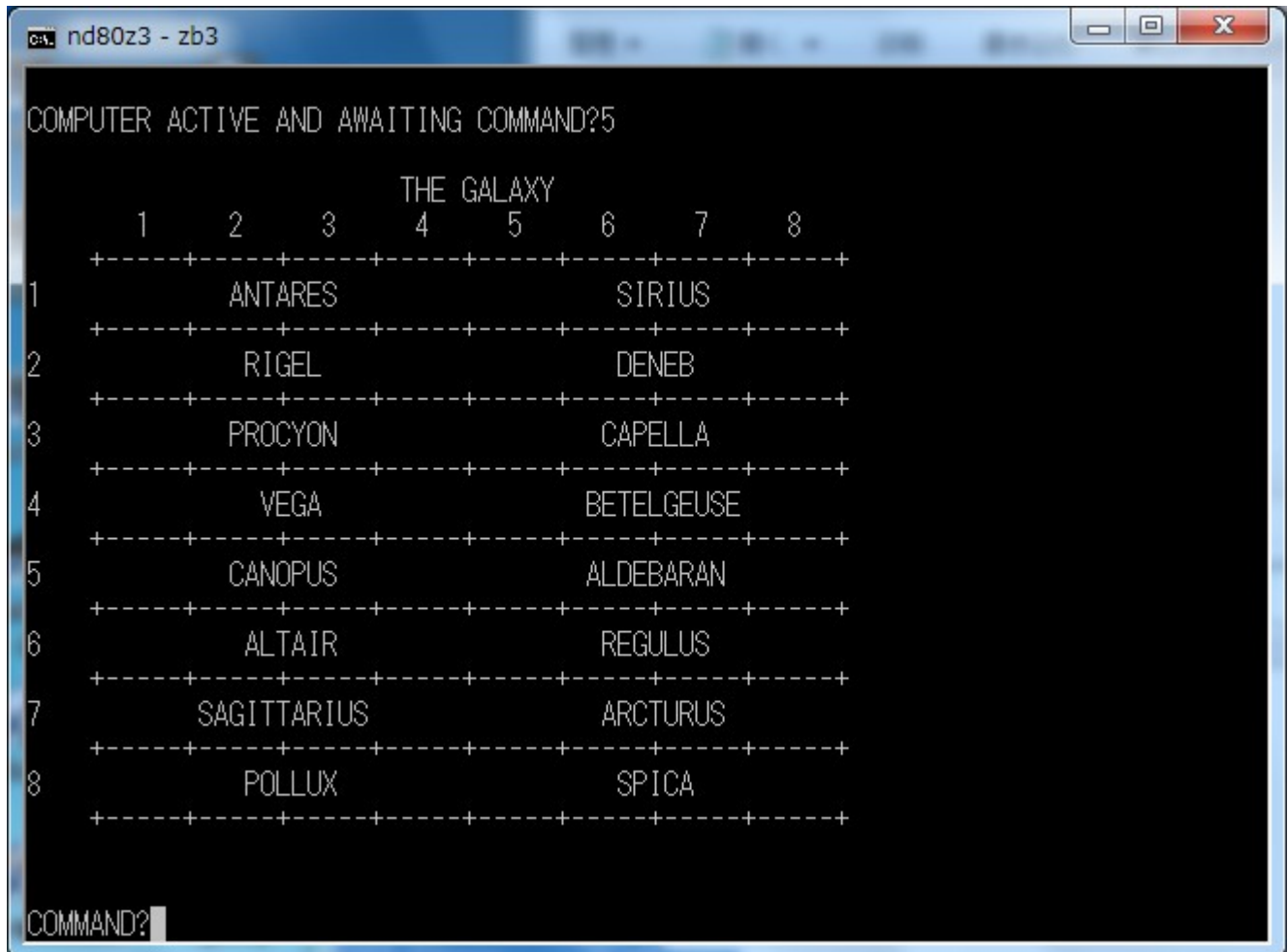
自分の位置から、敵位置の方向（DIRECTION）を計算して、攻撃します。

この計算について、わかったことがあります。

QUADRANTとはその綴りからも連想されますように4分円とか4分座標とかと訳されます。

「BASIC COMPUTER GAMES」のサイトの説明を読みましたら、その各小宇宙は全銀河宇宙を8×8に分割したときの、全宇宙内の位置を示す座標値であらわされるほかに、それぞれに固有の名前がつけられていることがわかりました。

その名前は、COMコマンドで数値5を入力すると画面に表示されます。



さらに同じ名前のついたQUADRANTは、それぞれ左からI、II、III、IVのローマ数字が付加されて区別されます。

たとえば前回図で説明しましたQUADRANT（6，7）はREGULUSⅢになります。

この名前は前回お見せしましたゲーム開始画面にも表示されています（下の画面中ほどに表示されています）。

```
nd80z3 - zb3
DESTRUCT THE 13 KLINGON WARSHIPS WHICH HAVE INVADED
THE GALAXY BEFORE THEY CAN ATTACK FEDERATION HEADQUARTERS.
THIS GIVES YOU 25 DAYS. THERE IS
1 STARBASE IN THE GALAXY FOR RESUPPLYING YOUR SHIP.

ARE YOU READY TO ACCEPT COMMAND ('N' FOR INSTRUCTIONS)?y

YOUR MISSION BEGINS WITH YOUR STARSHIP LOCATED
IN THE GALACTIC QUADRANT, 'REGULUS III'.

      COMBAT AREA      CONDITION RED
      SHIELDS DANGEROUSLY LOW

+--1---2---3---4---5---6---7---8--+
1 |          <E>          | 1      STARDATE          2000.0
2 |          |          | 2      CONDITION          *RED*
3 |   *          |          | 3      QUADRANT          6,7
4 |          |          *   | 4      SECTOR          1,2
5 |          |          | 5      PHOTON TORPEDOES      10
6 |   +K+          |          | 6      TOTAL ENERGY      3000
7 |   *          |          | 7      SHIELDS          0
8 |          |          | 8      KLINGONS REMAINING    13
+--1---2---3---4---5---6---7---8--+

COMMAND? 
```

ゲームの進行は全てCOMMAND?に対してコマンドを入力することで行なわれます。  
コマンドは以下のものがあります (全て大文字で入力します)。  
COMMAND?に対してコマンド以外の文字を入力すると、下のように全コマンドが表示されます。

```
c:\nd80z3 - zb3dos1t
 2 |          * | 2 |   CONDITION   GREEN
 3 |          | 3 |   QUADRANT    1,1
 4 |          | 4 |   SECTOR      1,1
 5 |          | 5 | PHOTON TORPEDOES 10
 6 |          | 6 | TOTAL ENERGY 3000
 7 |          | 7 | SHIELDS        0
 8 |          | 8 | KLINGONS REMAINING 8
  +--1--2--3--4--5--6--7--8--+

COMMAND?a

ENTER ONE OF THE FOLLOWING:
-----
NAV (TO SET COURSE)
SRS (FOR SHORT RANGE SENSOR SCAN)
LRS (FOR LONG RANGE SENSOR SCAN)
PHA (TO FIRE PHASERS)
TOR (TO FIRE PHOTON TORPEDOES)
SHE (TO RAISE OR LOWER SHIELDS)
DAM (FOR DAMAGE CONTROL REPORTS)
COM (TO CALL ON LIBRARY-COMPUTER)
XXX (TO RESIGN YOUR COMMAND)

COMMAND?
```

NAVはエンタープライズ号をワープさせる命令です（後述）。  
SRS（ショートレンジセンサースキャン）は今いるQUADRANT内の情報をサーチして示します。  
QUADRANTの座標画面とともに右側に基本的な情報が表示されます。

```
c:\nd80z3 - zb3dos1t
NAV (TO SET COURSE)
SRS (FOR SHORT RANGE SENSOR SCAN)
LRS (FOR LONG RANGE SENSOR SCAN)
PHA (TO FIRE PHASERS)
TOR (TO FIRE PHOTON TORPEDOES)
SHE (TO RAISE OR LOWER SHIELDS)
DAM (FOR DAMAGE CONTROL REPORTS)
COM (TO CALL ON LIBRARY-COMPUTER)
XXX (TO RESIGN YOUR COMMAND)

COMMAND?SRS

  +--1--2--3--4--5--6--7--8--+
1 | <E>      *                    | 1   STARDATE           2000.0
2 |          *                    | 2   CONDITION        GREEN
3 |          *                    | 3   QUADRANT         1,1
4 |          *                    | 4   SECTOR           1,1
5 |          *                    | 5   PHOTON TORPEDOES  10
6 |          *                    | 6   TOTAL ENERGY    3000
7 |          *                    | 7   SHIELDS          0
8 |          *                    | 8   KLINGONS REMAINING 8
  +--1--2--3--4--5--6--7--8--+

COMMAND?
```

LRS（ロングレンジセンサースキャン）は今いるQUADRANTを中心にしてその回りの9個のQUADRANTの情報を示します。

```

c:\ nd80z3 - zb3dos1t
+--1--2--3--4--5--6--7--8--+
1 | <E>      *          | 1   STARDATE           2000.0
2 |          *          | 2   CONDITION        GREEN
3 |          |          | 3   QUADRANT          1,1
4 |          |          | 4   SECTOR            1,1
5 |          |          | 5   PHOTON TORPEDOES   10
6 |          |          | 6   TOTAL ENERGY    3000
7 |          |          | 7   SHIELDS             0
8 |          |          | 8   KLINGONS REMAINING  8
+--1--2--3--4--5--6--7--8--+

COMMAND?LRS

LONG RANGE SCAN FOR QUADRANT 1,1

-----
| *** | *** | *** |
-----
| *** | 002 | 004 |
-----
| *** | 008 | 004 |
-----

COMMAND?

```

今いるQUADRANTは銀河内座標（1， 1）で、銀河の左上隅ですから、上と左のQUADRANTの情報はありません（\*\*\*が表示されます）。

3桁の情報はそのQUADRANTの状態を示します。

1位の数は含まれる星の数を示します。

中央に表示されているのが現在いるQUADRANTで、上の座標画面を見て分かるように星（\*）が2つあります。

10位の数はSTARBASE（宇宙基地）を示します。

100位の数は敵（KLINGON）の数を示します。

この範囲には敵はいませんから別のQUADRANTに移動して敵を探します。

移動するにはNAVコマンドを使います。

```

c:\nd80z3 - zb3dos1t
-----
COMMAND?NAV
COURSE (0-9)?8.33333
WARP FACTOR (0-8)?3.1622

NOW ENTERING PROCYON IV QUADRANT . . .

      COMBAT AREA      CONDITION RED
      SHIELDS DANGEROUSLY LOW

+--1--2--3--4--5--6--7--8--+
1 |          <E>          | 1   STARDATE           2001.0
2 |                      | 2   CONDITION          *RED*
3 |                      | 3   QUADRANT           3,4
4 |                      | 4   SECTOR            1,2
5 |                      | 5   PHOTON TORPEDOES  10
6 |                      | 6   TOTAL ENERGY    2965
7 |                      | 7   SHIELDS           0
8 |                      | 8   KLINGONS REMAINING 8
+--1--2--3--4--5--6--7--8--+

COMMAND?

```

(2, 4)に移動するつもりでCOURSE (方向)に8. 66667と入力するところをうっかりして8. 33333と入力してしまいました。

そのため1つ下のQUADRANT (3, 4)に行ってしまいました。

でも怪我の功名で敵 (+K+) を発見しました。

戦闘モードが発令されCONDITIONは\*RED\*になりました (通常はGREENです)。

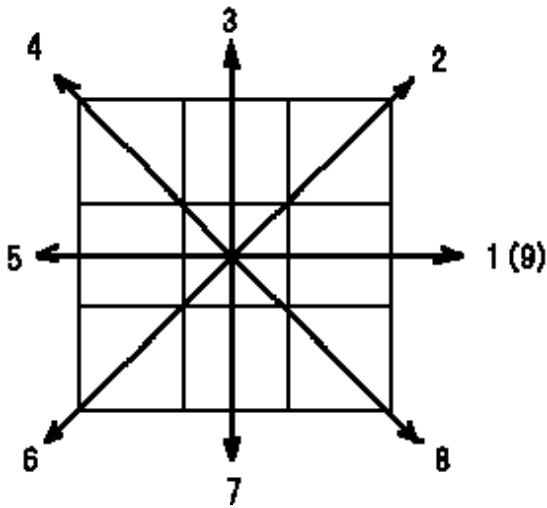
シールドがしてありませんから、SHIELDS DANGEROUSLY LOWと表示されています。

さてまずはQUADRANT間移動について説明します。

移動する方向は図のように1~8の数値で示されます。

図の(9)は8~1の間の方向を示すときに仮に使う数です。





図のように水平、垂直、斜め45度の方向は整数で示されます。

画面は横長表示ですが計算では図のように正方形と考えます。

さて問題は上の図のいずれでもない方向（方角ではありません！）の数値での示し方です。

ちょっと見ると45度が”1”ですから三角関数のTANのように思えます。

STARTREK ゲームの日本語の説明があるサイト [http://www.yosyisland.com/Page/Yosy\\_StarTrek\\_Form.aspx](http://www.yosyisland.com/Page/Yosy_StarTrek_Form.aspx) の管理者のかたもそのように書いてみえます。

逆三角関数を計算して…と書いていらっしゃるようですが、おそらく思い違いです。

そういうバージョンもあったかもしれませんが、おそらくそんなレベルのものはなかったのでは？

ちなみにMBASICにはATN（アークタンジェント、逆正接）もありますが、STARTREKでは使われていません。

この方向は敵の方向を計算するときにも使われていてCOMコマンドで得ることができますが、上記サイトの管理者様はCOMで得られる数値は不正確と書いていらっしゃいます。

三角関数で計算されているという仮定での批評ですがCOMの計算結果は正確です。

実はSTARTREKでは三角関数などは使われていません。

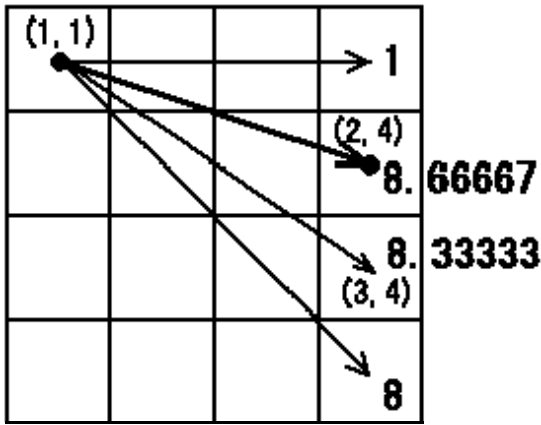
方向の算出は結構シンプルな考え方で算出しているようです。

いくつかのケースで確認してみましたがそれに間違いはないと思います。

ゲームの進行に合わせて説明をいたしますが、まずは今回のケースについて説明をします。

本当は（1， 1）から（2， 4）に移動するつもりでした。

するとそれは下の図のようになります。



下へ1ポイント右へ3ポイントの移動です。

下へ3、右へも3の移動方向は8で、右へ3の移動方向は1（9）ですから、（1，1）から（2，4）への移動方向はその間の数値です。

その間は3等分して考えます。

すると図のように8.66667という数値が得られます。

そのようにするところを間違えて8.33333と入力してしまったものですから、ひとつ下の（2，4）に行ってしまいました。

ところでWARP FACTOR（ワープ距離）ですが、これはQUADRANT間の水平垂直移動距離を1とします。

QUADRANT内の小移動は1未満の少数値になります。

今回は斜め方向への移動です。

この場合にはどのようにに計算するのでしょうか？

おわかりですね。

そうです。中学校で習ったピタゴラスの定理（三平方の定理）で計算をします。

下に1、右に3ですから、その距離は $1^2+3^2=10$ の平方根3.1622になります。

しかし実際には下に2、右に3の移動方向を指定してしまいましたから、その移動距離は $2^2+3^2=13$ の平方根3.6056が必要でした。

それでもなんとか（2，4）にたどりつきましたが元いた（1，1）内の位置からは少しずれてしまいました。

COMコマンドはエンタープライズ号のコンピュータに情報を要求するコマンドです。

COMMAND？

のところにCOM [Enter]

と入力すると

COMPUTER ACTIVE AND AWAITING COMMAND？

と表示されます。

ここには0～5を入力しますが、それよりも大きな数を入力すると下の画面のように各サブコマンドの説明が表示されたあと、再入力を要求されます。

```
c:\nd80z3 - zb3dos1t
>goto 7360
FUNCTIONS AVAILABLE FROM LIBRARY-COMPUTER:

ERR:31
 7365 PRINT "-----";
>
>LIST 7365
 7366 PRINT "-----":PRINT
>go to 7360

ERR:23
>goto 7320
COMPUTER ACTIVE AND AWAITING COMMAND?9

FUNCTIONS AVAILABLE FROM LIBRARY-COMPUTER:
-----

 0 = CUMULATIVE GALTIC RECORD
 1 = STATUS REPORT
 2 = PHOTON TORPEDO DATA
 3 = STARBASE NAV DATA
 4 = DIRECTION/DISTANCE CALCULATOR
 5 = GALAXY 'REGION NAME' MAP

COMPUTER ACTIVE AND AWAITING COMMAND?
```

とりあえずここでは2を使います。

2は敵に光子魚雷を発射するときの方位データを計算してくれます。

```
c:\nd80z3 - zb3dos1t
>LIST 7365
 7366 PRINT "-----":PRINT
>go to 7360

ERR:23
>goto 7320
COMPUTER ACTIVE AND AWAITING COMMAND?9

FUNCTIONS AVAILABLE FROM LIBRARY-COMPUTER:
-----
 0 = CUMULATIVE GALTIC RECORD
 1 = STATUS REPORT
 2 = PHOTON TORPEDO DATA
 3 = STARBASE NAV DATA
 4 = DIRECTION/DISTANCE CALCULATOR
 5 = GALAXY 'REGION NAME' MAP

COMPUTER ACTIVE AND AWAITING COMMAND?2

FROM ENTERPRISE TO KLINGON BATTLE CRUSER
DIRECTION =8.16667
DISTANCE =7.81025

COMMAND?|
```

DIRCTION=8. 16667

DISTANCE=7. 81025

と答えてくれました。

光子魚雷で敵を攻撃するのにDISTAMCEは必要ありません。

DIRCTION（発射方向）だけが必要な情報です。

あ。

ここは前回の敵と遭遇したときの場面の続きです。

こちらが前回のその画面です。

```
nd80z3 - zb3dos1t
-----
COMMAND?NAV
COURSE (0-9)?8.33333
WARP FACTOR (0-8)?3.1622

NOW ENTERING PROCYON IV QUADRANT . . .

      COMBAT AREA      CONDITION RED
      SHIELDS DANGEROUSLY LOW

+---1---2---3---4---5---6---7---8--+
1 |          <E>          | 1      STARDATE           2001.0
2 |          |          | 2      CONDITION           *RED*
3 |          |          | 3      QUADRANT            3,4
4 |          |          | 4      SECTOR              1,2
5 |          |          *  | 5      PHOTON TORPEDOES      10
6 |          |          +K+ | 6      TOTAL ENERGY       2965
7 |          |          | 7      SHIELDS                0
8 |          |          *  | 8      KLINGONS REMAINING     8
+---1---2---3---4---5---6---7---8--+

COMMAND?
```

前回も書きましたように実は敵までの方向は、COMPUTERに問い合わせなくても簡単に求めることができます。

上の画面でエンタープライズ号は（1， 2）にいます。

敵は（6， 8）です。

すると敵までは、右に6、下に5移動することになります。

右に6、下に6の方向が” 8” ですから、それよりも1／6だけ上の位置です。

つまり $8 + 1/6 = 8.16667$ になります。

光子魚雷の発射コマンドはTORです。

TOR [Enter] と入力しました。

```
c:\ nd80z3 - zb3dos1t
3 = STARBASE NAV DATA
4 = DIRECTION/DISTANCE CALCULATOR
5 = GALAXY 'REGION NAME' MAP

COMPUTER ACTIVE AND AWAITING COMMAND?2

FROM ENTERPRISE TO KLINGON BATTLE CRUSER
DIRECTION =8.16667
DISTANCE =7.81025

COMMAND?TOR

PHOTON TORPEDO COURSE (1-9)?8.16667
TORPEDO TRACK:
      2,3
      3,4
      3,5
      4,6
      5,7
      6,8

*** KLINGON DESTROYED ***

COMMAND?
```

おお。  
命中です。  
KLINGONを破壊しました。

#### ●宇宙基地にドッキング

やっと宇宙基地を見つけました。  
いよいよこれから今度は宇宙基地とドッキングです。

```

NOW ENTERING SIRIUS III QUADRANT . . .

  +--1---2---3---4---5---6---7---8--+
  1 |          *                          | 1   STARDATE           2002.0
  2 |          >B<                          | 2   CONDITION          GREEN
  3 |   *                                     | 3   QUADRANT           1,7
  4 |          *                          | 4   SECTOR             8,3
  5 |          |                          | 5   PHOTON TORPEDOES   9
  6 |          *                          | 6   TOTAL ENERGY     2928
  7 |          *                          | 7   SHIELDS           1000
  8 |          <E>                          | 8   KLINGONS REMAINING 7
  +--1---2---3---4---5---6---7---8--+

COMMAND?COM

COMPUTER ACTIVE AND AWAITING COMMAND?3

FROM ENTERPRISE TO STARBASE:
DIRECTION =2.33333
DISTANCE =7.2111

COMMAND?NAV

COURSE (0-9)?2.33333
WARP FACTOR (0-8)?0.72111

```

先ほどと同じように宇宙基地までの方位と距離をコンピュータに聞きます。

COMMAND?

にCOM [Enter] と入力して、次に

3 [Enter] と入力します。

DIRECTION=2.33333

DISTANCE=7.2111

というデータが得られました。

これも実はCOMPUTERに聞かなくても計算で求めることができます。

エンタープライズ号は(8, 3)にいます。

宇宙基地は(2, 7)です。

上に6、右に4行ったところです。

上に6、右に6が”2”の方向です。

それよりも2/6だけ”3”の方向ですから、

方向は $2 + 2/6 = 2.33333$ になります。

そして距離は $6^2 + 4^2 = 52$ の平方根 $7.2111$ になります。

そこでNAVコマンドでこのデータを入力するのですが、ただしQUADRANT内の移動距離は計算値の1/10を入力します。

宇宙基地にドッキングできました。

```
DIRECTION =2.33333
DISTANCE =7.2111

COMMAND?NAV

COURSE (0-9)?2.33333
WARP FACTOR (0-8)?0.72111

WARP ENGINES SHUT DOWN AT SECTOR3,6 DUE TO BAD NAVAGATION
SHIELDS DROPPED FOR DOCKING PURPOSES.
  +--1---2---3---4---5---6---7---8-->
1 |          *          | 1   STARDATE           2002.7
2 |          >B<        | 2   CONDITION          DOCKED
3 | *              <E> | 3   QUADRANT           1,7
4 |          *          | 4   SECTOR             3,6
5 |          |          | 5   PHOTON TORPEDOES      10
6 |          *          | 6   TOTAL ENERGY        3000
7 |          *          | 7   SHIELDS              0
8 |          |          | 8   KLINGONS REMAINING    7
  +--1---2---3---4---5---6---7---8--+
```

ERR:15  
11824 2111

CONDITIONのところ DOCKEDと表示されています。  
前回敵 (KLINGON) を光子魚雷で破壊したときの画面についてです。

まずはこちらの画面です。



```
-----
COMMAND?NAV
COURSE (0-9)?8.33333
WARP FACTOR (0-8)?3.1622

NOW ENTERING PROCYON IV QUADRANT . . .

      COMBAT AREA      CONDITION RED
      SHIELDS DANGEROUSLY LOW

+--1--2--3--4--5--6--7--8--+
1 |          <E>          | 1
2 |          |          | 2
3 |          |          | 3
4 |          |          | 4
5 |          |          * | 5
6 |          |          +K+ | 6
7 |          |          | 7
8 |          |          * | 8
+--1--2--3--4--5--6--7--8--+

STARDATE          2001.0
CONDITION          *RED*
QUADRANT           3,4
SECTOR            1,2
PHOTON TORPEDOES  10
TOTAL ENERGY     2965
SHIELDS           0
KLINGONS REMAINING 8

COMMAND?
```

よく見ますとエンタープライズ号と敵艦（KLINGON）との間に星（\*）があります。

光子魚雷が星に当たってしまうと敵を破壊できません。

むむ。

やばいなあ。

と思いつつ、発射したのです。

そしてこちらの画面です。

```
c:\nd80z3 - zb3dos1t
3 = STARBASE NAV DATA
4 = DIRECTION/DISTANCE CALCULATOR
5 = GALAXY 'REGION NAME' MAP

COMPUTER ACTIVE AND AWAITING COMMAND?2

FROM ENTERPRISE TO KLINGON BATTLE CRUSER
DIRECTION =8.16667
DISTANCE =7.81025

COMMAND?TOR

PHOTON TORPEDO COURSE (1-9)?8.16667
TORPEDO TRACK:
      2,3
      3,4
      3,5
      4,6
      5,7
      6,8

*** KLINGON DESTROYED ***

COMMAND?
```

命中までの光子魚雷のコースが座標値で示されています。  
ここはいまどきのディスプレイゲームでしたら光子魚雷が進む航跡が表示されるところでしょうが、文字のスクロール表示しかできない昔のコンピューターゲームですから、数値しか表示されません。  
しかしここは潜水艦の乗組員がレシーバーを耳に当てながら敵まであと〇〇メートル！と言っている場面などと同じようなシーンを連想したりするとそれはそれで楽しいではありませんでしょうか。  
で。  
この光子魚雷のコースなのですが。

まずはこちらをご覧ください。  
エンタープライズ号（E）と敵（K）の位置です。

	1	2	3	4	5	6	7	8
1		E						
2								
3								
4								
5						*		
6								K
7								
8								

間の星（\*）はまさに光子魚雷のコース上にあるように見えませんか？  
なぜ星に当たらなかったのでしょうか。

その疑問に対する答えは、光子魚雷のコースを上座標にプロットしてみるとわかります。

	1	2	3	4	5	6	7	8
1		E						
2			●					
3				●	●			
4						●		
5						*	●	
6								K
7								
8								

まさにデジタルの線分描画そのものです。

（3， 5）でコースが1ポイント右にずれたお蔭でかろうじて星を避けることができたのです。