

ミサイル弾道の計算 (準備1 と着弾時間)

発射地点と着弾地点の距離と半射程角の計算

point 1 : 39.16022460835828N , 127.4456701811123E

point 2 : 13.43333N , 144.75E

```
/* 弾道計算の準備1 */
```

```
/* 緯度経度から2地点間距離と半射程角を計算 */
```

```
/* Calculate distance and half-range angle from latitude and longitude of 2 points -  
Сейджиро Кусафуса */
```

```
#include <stdio.h>
```

```
#include <math.h>
```

```
/* 数学定数 math constants */
```

```
#define PI 3.141592653589793 /* 円周率 Pi */
```

```
#define e 2.718281828459045 /* Napier number */
```

```
/* 物理定数 */
```

```
#define E 5.972e24 /* kg, 地球の質量 Mass of the earth */
```

```
#define R 6.36e6 /* m, 地球の半径 radius at the equator */
```

```
/* 変数の定義 */
```

```
double lat1,lat2; // latitude of each point, degree
```

```
double lr1,lr2; // radian of ditto
```

```
double long1,long2; // longitude of each point, degree
```

```
double gr1,gr2; // radian
```

```
double h1,h2; // altitude of each point m
```

```
//double a=6378136.0; // radius at the equator m
```

```
double e2=0.006694470; // squared eccentricity
```

```
double x1,y1,z1, x2,y2,z2; // each dimension on Cartesian coordinate m
```

```
double r; // direct distance between two points m
```

```
double s; // ground surface distance between two points m
```

```
double w; // half range angle between two points, i.e., 1/2 of the center angle
```

```
double wr; // radian of ditto
```

```
double rad; // PI/180
```

```
double N1,N2; // latitude corrected earth radius
```

```
int main(void)
```

```
{
```

```
    rad=PI/180.0;
```

```
    h1=h2=0.0;
```

```
//地点1 緯度経度 latitude & longitude at point 1
```

```
    scanf("%lf %lf",&lat1,&long1);
```

```
    if(long1<0) long1=360.0+long1;
```

```
    lr1=lat1*rad; gr1=long1*rad;
```

```

//地点2 緯度経度 latitude & longitude at point 2
scanf("%lf %lf",&lat2,&long2);
if(long2<0) long2=360.0+long2;
lr2=lat2*rad; gr2=long2*rad;

// 直交座標値の計算 calculation of each dimension on Cartesian coordinate
N1=R/(sqrt(1.0-e2*sin(lr1)*sin(lr1)));
x1=(N1+h1)*cos(lr1)*cos(gr1);
y1=(N1+h1)*cos(lr1)*sin(gr1);
z1=(N1*(1.0-e2)+h1)*sin(lr1);

N2=R/(sqrt(1.0-e2*sin(lr2)*sin(lr2)));
x2=(N2+h2)*cos(lr2)*cos(gr2);
y2=(N2+h2)*cos(lr2)*sin(gr2);
z2=(N2*(1.0-e2)+h2)*sin(lr2);

//計算結果 results of the calculation
r=sqrt((x1-x2)*(x1-x2)+(y1-y2)*(y1-y2)+(z1-z2)*(z1-z2)); /* 直距離 */
wr=asin(r/2/R); /* 半射程角rad */
w=wr/rad;
s=R*2*wr; /* 地表面距離 */

putchar('\n');
printf("地点1の緯度 : %lf 経度 : %lf\n",lat1,long1);
printf("地点2の緯度 : %lf 経度 : %lf\n",lat2,long2);
printf("直距離 : %9.3lf km\n",r/1000);
printf("地表面距離 : %9.3lf km\n",s/1000);
printf("半射程角 : %8.4lf °\n",w);
getchar(); getchar();
return 0;
}

```

計算結果:

地点1の緯度 : 39.160225 経度 : 127.445670

地点2の緯度 : 13.433333 経度 : 144.750000

直距離 : 3272.740 km

地表面距離 : 3309.968 km

半射程角 : 14.9094 °

さらに、上記の2点間を発射点と着弾点としてミサイルを発射したときの飛行時間を計算する。

```
/* 弾道計算 飛行時間 */  
/* 入力：半射程角、長径、離心率 */  
/* 出力：飛翔距離、平均速度、飛翔時間 */  
/* MKS単位系 */
```

```
#include <stdio.h>  
#include <math.h>  
#define PI 3.14159265358979  
#define R 6378136 /* 赤道面平均半径 m */  
#define U (3.98602e+14) /* 地心引力係数 */  
#define N 100 /* 降弧の分割数 */
```

```
int main(void) {  
    double a; /* 楕円軌道長径 */  
    double e; /* 離心率 */  
    double r; /* 動径 */  
    double v; /* ミサイル速度 m/s */  
    double vm; /* ミサイル平均速度 m/s */  
    double vs; /* ミサイル速度の加算値 */  
    double w; /* 半射程角 rad */  
    double dw; /* 分割半射程角 w/N */  
    double rad; /* 度→ラジアン変換係数 */  
    double ds,dx,dy; /* 弾道経路の微少分 */  
    double s; /* 弾道距離 m */  
    double tf; /* 飛翔時間 s */  
    int hh,mm; /* 時 分 */  
    double ss; /* 秒 */  
    double tmp1,tmp2; /* working variable */  
    int i;  
    rad=PI/180.0;  
  
    scanf("%lf %lf %lf",&w,&a,&e);  
    printf("半射程角 = %4.1lf °\n",w);
```

```

printf("長径 = %6.1lf Km\n",a);
printf("離心率 = %6.4lf\n\n",e);
w=w*rad;
dw=w/N;
a=a*1000;
vs=0.0;
ds=0.0;
printf(" 角度   動径km   速度m/s\n");
for (i=0;i<=N;i++){
    r=a*(1.0-e*e)/(1.0-e*cos(dw*i));
    v=sqrt(U*((2.0/r)-(1.0/a)));
    vs+=v;
    if(i==0){
        tmp1=r;
    }
    else{
        tmp2=r;
    }
    if(i>1){
        dx=tmp2*sin(dw);
        dy=tmp1-tmp2*cos(dw);
        ds+=sqrt(dx*dx+dy*dy);
        tmp1=tmp2;
    }
    if(i%10==0){
        /* 100 分割中の 10 データを抽出表示する */
        printf(" %8.5lf %6.1lf %6.1lf\n",dw*i/rad,r/1000,v);
    }
}
s=ds*2.0; /* 飛翔距離 (昇弧 + 降弧) */
vm=vs/(N+1); /* 平均速度 */
tf=s/vm; /* 飛翔時間 (秒) */
hh=(int)(tf/3600); /* 秒→時 : 分 : 秒 変換 */
mm=(int)((tf-hh*3600)/60.0);
ss=tf-hh*3600-mm*60;
printf("\n 平均速度 = %6.1lf m/s\n",vm);
printf("\n 飛翔距離 = %6.1lf km\n",s/1000);
printf("\n 飛翔時間 = %6.1lf s <%2d:%2d:%4.1lf>\n",tf,hh,mm,ss);
getchar();getchar();
return 0;
}

```

計算結果:

半射程角 = 14.9 °

長徑 = 4018.0 Km

離心率 = 0.7800

角度	動徑km	速度m/s
0.00000	7152.0	3501.6
1.49094	7143.5	3520.6
2.98188	7117.9	3577.2
4.47282	7075.6	3669.4
5.96376	7017.4	3794.7
7.45470	6944.0	3949.9
8.94564	6856.4	4131.4
10.43658	6755.8	4335.8
11.92752	6643.5	4560.0
13.41846	6520.9	4801.0
14.90940	6389.4	5056.3

平均速度 = 4061.9 m/s

飛翔距離 = 3936.2 km

飛翔時間 = 969.1 s < 0:16: 9.1 >

```
半射程角 = 14.9 °
長徑 = 4018.0 Km
離心率 = 0.7800

  角度    動徑km    速度m/s
0.00000  7152.0  3501.6
1.49094  7143.5  3520.6
2.98188  7117.9  3577.2
4.47282  7075.6  3669.4
5.96376  7017.4  3794.7
7.45470  6944.0  3949.9
8.94564  6856.4  4131.4
10.43658 6755.8  4335.8
11.92752 6643.5  4560.0
13.41846 6520.9  4801.0
14.90940 6389.4  5056.3

平均速度 = 4061.9 m/s

飛翔距離 = 3936.2 km

飛翔時間 = 969.1 s < 0:16: 9.1 >
```

半射程角 = 14.9 °

長径 = 4172.0(Km)

離心率 = 0.5976

角度	動径 km	速度 m/s
0.00000	6665.2	4905.6
1.49094	6661.8	4911.7
2.98188	6651.8	4930.1
4.47282	6635.2	4960.4
5.96376	6612.0	5002.6
7.45470	6582.6	5056.3
8.94564	6546.9	5121.1
10.43658	6505.4	5196.5
11.92752	6458.1	5282.0
13.41846	6405.5	5377.2
14.90940	6347.8	5481.3

平均速度 = 5103.2 m/s

飛翔距離 = 3454.4 km

飛翔時間 = 676.9 s < 0:11:16.9 >

参照: 弾道計算 <http://imetrics.co.jp/mathematics/OrbitalMissile.pdf>