

すべてのファイルを <https://github.com/meowwowcat/report3.git> に置きました。

課題 20

方法

```
1  #include<stdio.h>
2  #include<math.h>
3  #define NDAT 21
4  #define dv 0.001
5  #define PRECISION 0.1
6  #define vmin 4.00003
7
8  float func(float x,float v,float x0);
9  float calc_chi2(float [],float x[],float s[],float a,float b );
10 float dcdv(float t[],float x[],float s[],float a,float b);
11
12 int main(void){
13     FILE*fp;
14     float t[NDAT],x[NDAT],s[NDAT],a,b;
15     int ih;
16     if((fp=fopen("u.dat","r"))!=NULL){
17         for(ih=0;ih<NDAT;ih++){
18             fscanf(fp,"%f %f %f %f %f",&t[ih],&x[ih],&a,&b,&s[ih]);
19             s[ih]=PRECISION;
20         }
21         fclose(fp);
22     }
23     float par_v=2.2, par_x0=3.4;
24     float d2cdv2,err;
25     d2cdv2=(dcdv(t,x,s,vmin+dv,par_x0)-dcdv(t,x,s,vmin,par_x0))/dv;
26     err=sqrt(2/d2cdv2);
27     printf("Solution = %f +- %f\n",vmin,err-1);
28     return 0;
29 }
30
31 float dcdv(float t[],float x[],float s[],float a,float b)
32 {
33     float dcdv=0;
34     dcdv=(calc_chi2(t,x,s,a+dv,b)-calc_chi2(t,x,s,a,b))/dv;
35     return dcdv;
36 }
37
38 float calc_chi2(float t[],float x[],float s[],float a,float b)
39 {
40     float chi2;
41     int i;
42     for(i=0;i<NDAT;i++){
43         chi2+=(x[i]-func(t[i],a,b))* (x[i]-func(t[i],a,b)) / (s[i] *
44 s[i]);
45     }
46     return chi2;
47 }
48
49 float func(float t,float v,float x0)
50 {
51     float x=v*t + x0;
52     return x;
```

3～10行目で文字の定義をする。16行目から22行目で課題16のファイルの読み込み。31行目で導関数を数値で計算。34行目のcalc_chi2で関数を計算。差分をとって近似。38行目から45行目で関数を計算。48以降で具体的な関数を定義。

結果

Solution = 4.000030 +- -0.999717

考察

課題16における v_0 にだいたい等しい。したがってこの値は妥当である。

課題 21

方法

```

1  #include<stdio.h>
2  #include<math.h>
3
4  #define NDAT 20
5  #define PRECISION 0.1
6
7  int main(void){
8      FILE*fp;
9      float
10     t[NDAT],x[NDAT],s[NDAT],y[NDAT],exp_y[NDAT],error_y[NDAT],err[NDAT];
11     float para_v,para_x0;
12     float err_v,err_x0;
13     int ih;
14
15     if((fp=fopen("u.dat","r"))!=NULL){
16         for(ih=0;ih<NDAT;ih++){
17             fscanf(fp,"%f %f %f %f %f",
18                 &t[ih],&x[ih],&y[ih],&exp_y[ih],&error_y[ih]);
19         }
20         fclose(fp);
21     }
22     else{
23         printf("FILE OPEN ERROR\n");
24         return -1;
25     }
26
27     for(ih = 0; ih < NDAT ; ih++){
28         s[ih]=PRECISION;
29     }
30
31     float SUM_T2 =0, SUM_T =0, SUM_X =0, SUM_TX =0,SUM_1 =0;
32     int i;
33     for(i=0;i<NDAT;i++){
34         SUM_T2 += t[i] * t[i] / (s[i] * s[i]);
35         SUM_T += t[i] / (s[i] * s[i]);
36         SUM_X += x[i] / (s[i] * s[i]);
37         SUM_TX += t[i] * x[i] / (s[i] * s[i]);
38         SUM_1 += 1.0 / (s[i] * s[i]);
39     }
40     para_v = (SUM_TX * SUM_1 - SUM_X * SUM_T) / (SUM_T2 * SUM_1 - SUM_T *
SUM_T );

```

```

39     para_x0 = (SUM_T2 * SUM_X - SUM_TX * SUM_T) / (SUM_T2 * SUM_1 - SUM_T *
SUM_T);
40
41     err_v = sqrt(SUM_1 / (SUM_T2 * SUM_1 - SUM_T * SUM_T));
42     err_x0 = sqrt(SUM_T2 / (SUM_T2 * SUM_1 - SUM_T * SUM_T));
43
44     printf("v = %f +- %f, x0 = %f +-%f\n", para_v, err_v, para_x0, err_x0);
45
46     return 0;
47 }

```

最小二乗法の計算に用いる色々な和の計算をする..

結果

$v = 3.999732 \pm 0.038778$, $x_0 = 10.000156 \pm 0.043095$

考察

課題 16 で用いた, v_x, x_0 の値とだいたい等しい。したがってこの値は妥当である。