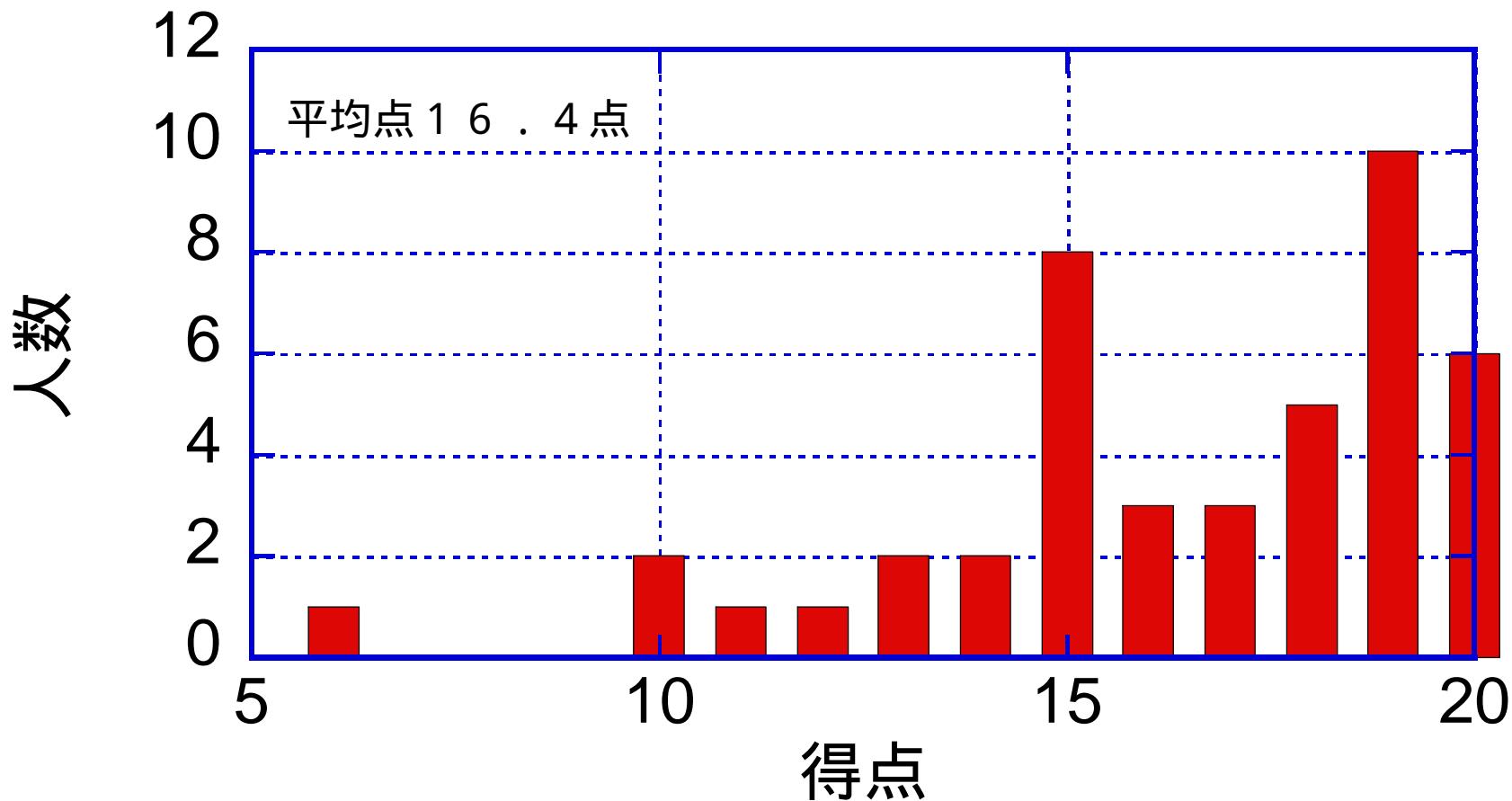


第1回中間試験 得点分布



12点以下の5名は、
本日3時30分以降に
復習用課題を解くこと

Step 18 配列（1次元）

- ベクトルや行列と同じ
- 変数宣言

```
int DATA[5];  
DATA[0], DATA[1],  
DATA[2], DATA[3],  
DATA[4] の5成分をもつ
```

- データ入力、表示

方法1 DATA[0]=1;
各成分を別々の変数として
取り扱う

方法2（右例参照）

```
01 #include <stdio.h>  
02 main()  
03 {  
04     int DATA[5];  
05     Int I;  
06     for(i=0;i<5;i++) {  
07         DATA[i]=i;  
08         printf("%d\n", DATA[i]);  
09     }
```

Step 18 配列（1次元） 初期化とコピー

```
/* sample18-3.c */
#include <stdio.h>
main()
{
    int a[5]=(0,1,2,3,4); ←———— 初期値をベクトル形式で代入可能
    int b[5];
    int i;
    for(i=0;i<5;i++) {
        b[i]=a[i];
    }
    for(i=0;i<5;i++) {
        printf("%d %d %d\n", i, a[i], b[i]);
    }
}
```

課題 1 (ex18-2.c)

- (1) 要素数360の配列を作り ,
- (2) 要素番号iの成分に , $\sin(i \deg)$ の値を代入し ,
- (3) $i = 0$ から359までの $\sin(i \deg)$ の値および
それまでの $\sin(i \deg)$ の総和を計算し ,
- (4) 途中経過を表示する プログラムを作れ .

Step 18 配列（1次元）応用&参考

配列要素へのキーボード入力（各要素は独立変数）

課題 2 (app18.c)

2つの3次元ベクトル **a**, **b** について、

- (1) 成分をキーボードから入力し、
- (2) **a** と **b** の内積を計算し、
- (3) 計算結果を表示する

プログラムを作れ

<内積の計算>

$$\mathbf{a} = (a[0], a[1], a[2])$$

$$\mathbf{b} = (b[0], b[1], b[2])$$

$$S = \sum_{i=0}^2 a[i] * b[i]$$

```
/* app18.c */
#include <stdio.h>
main()
{
    int a[3], b[3];
    int i, S=0;

    for(i=0; i<3; i++) {
        printf("a[%d]=", i);
        scanf("%d", &a[i]);
    }
    for(i=0; i<3; i++) {
        printf("b[%d]=", i);
        scanf("%d", &b[i]);
    }
    以下は自分で
}
```

Step 19 配列（多次元）

- 2次元配列 $b[2][2]$; 2行2列
成分は, $b[0][0], b[0][1], b[1][0], b[1][1]$
- 3次元以上の配列 $c[i][j][k]$ も定義可能

課題 3 (app19-1.c)

九九の表づくり

sample19-1.cを参考に,

右のフォーマットで九九の表を作れ

```
emerald.fml.mech.tuat.ac.jp> cc s:  
emerald.fml.mech.tuat.ac.jp> a.out  
1 2 3 4 5 6 7 8 9  
1 1 2 3 4 5 6 7 8 9  
2 2 4 6 8 10 12 14 16 18  
3 3 6 9 12 15 18 21 24 27  
4 4 8 12 16 20 24 28 32 36  
5 5 10 15 20 25 30 35 40 45  
6 6 12 18 24 30 36 42 48 54  
7 7 14 21 28 35 42 49 56 63  
8 8 16 24 32 40 48 56 64 72  
9 9 18 27 36 45 54 63 72 81  
emerald.fml.mech.tuat.ac.jp>
```

Step 19 配列（多次元）応用

課題 4 (app19-2.c)

2行2列の整数行列の行列式・逆行列を計算・表示しよう。

- (1) 入力：4成分（整数）をキーボード入力し、画面に表示。
- (2) 行列式・逆行列を計算し、以下の画面例に従って結果を表示。

逆行列がある場合

```
-----実行開始-----
a[0][0]の成分を入力 =>3
a[0][1]の成分を入力 =>5
a[1][0]の成分を入力 =>1
a[1][1]の成分を入力 =>2

行列aは 3 5
         1 2

行列式は 1
逆行列は 2.00 -5.00
           -1.00 3.00
-----おしまい-----
```

逆行列がない場合

```
-----実行開始-----
a[0][0]の成分を入力 =>2
a[0][1]の成分を入力 =>2
a[1][0]の成分を入力 =>2
a[1][1]の成分を入力 =>2

行列aは 2 2
         2 2

行列式の値は0, 逆行列はありません
-----おしまい-----
```

解答例 ex18-2.c

```
/* ex18-2.c */
#include <stdio.h>
#include <math.h>

main()
{
    float sinq[360];
    float pi=M_PI,sum;
    int i;

    sum=0.0;
    for(i=0;i<360;i++) {
        sinq[i]=sin((float)i/180.0*pi);
        sum=sum+sinq[i];
        printf("sin[%3d deg] = %f, sum = %f\n", i,sinq[i],sum);
    }
}
```

解答例 (app18.c)

```
/* app18.c */
#include <stdio.h>
main()
{
    int a[3],b[3];
    int i,S=0;

    for(i=0;i<3;i++) {
        printf("a[%d]=", i);
        scanf("%d", &a[i]);
    }
    for(i=0;i<3;i++) {
        printf("b[%d]=", i);
        scanf("%d", &b[i]);
    }

    for(i=0;i<3;i++) {
        S=S+a[i]*b[i];
    }

    printf("(a[0], a[1], a[2]) = (%d, %d, %d)\n", a[0], a[1], a[2]);
    printf("(b[0], b[1], b[2]) = (%d, %d, %d)\n", b[0], b[1], b[2]);
    printf("S = %d\n", S);
}
```

解答例(app19-1.c)

```
/* app19-1.c */
#include <stdio.h>

main()
{
    int m,n;
    int DATA[10][10];

    for(n=1;n<10;n++) {
        for(m=1;m<10;m++) {
            DATA[n][m]=m*n;
        }
    }

    for(m=1;m<10;m++) {
        DATA[0][m]=m;
    }
    for(n=1;n<10;n++) {
        DATA[n][0]=n;
    }

    for(n=0;n<10;n++) {
        for(m=0;m<10;m++) {
            if((n==0) && (m==0)) {
                printf("   ");
            }
            else {
                printf("%3d", DATA[n][m]);
            }
        }
        printf("\n");
    }
}
```

横1行目 →

縦1行目 →

左上端は空白 →

一行書き終えたら改行 →

解答例 app19-2.c

```
/* app19-2.c */
#include <stdio.h>
main()
{
    int    a[2][2],i,j,detA;
    float ai[2][2];

    for(i=0;i<=1;i++){
        for(j=0;j<=1;j++){
            printf("a[%d][%d]の成分を入力 => ", i, j);
            scanf("%d", &a[i][j]);
        }
    }

    printf("\n");
    printf("行列aは %3d %3d\n", a[0][0], a[0][1]);
    printf("          %3d %3d\n", a[1][0], a[1][1]);

    detA = a[0][0]*a[1][1] - a[1][0]*a[0][1];

    if(detA==0) {
        printf("\n");
        printf("行列式の値は%d, 逆行列はありません\n", detA);
    }
    else {
        ai[0][0] = (float)a[1][1]/(float)detA;
        ai[0][1] = -(float)a[0][1]/(float)detA;
        ai[1][0] = -(float)a[1][0]/(float)detA;
        ai[1][1] = (float)a[0][0]/(float)detA;
        printf("\n");
        printf("行列式は%7d\n", detA);
        printf("逆行列は %7.2f %7.2f\n", ai[0][0], ai[0][1]);
        printf("          %7.2f %7.2f\n", ai[1][0], ai[1][1]);
    }
}
```