

# 学年末試験解答用紙 (1E 情報処理基礎)

電気情報工学科

学籍番号

氏名

## 1 配列

[問 1] 3点

複数の同じ型のデータからなるデータ構造である。配列名と 0 以上の整数の添字—インデックス—を指定することにより、  
各々のデータにアクセスする。

[問 2] 2点

```
int hoge[1000];  
double fuga[99];
```

[問 3] 2点

```
int hoge[100][100];  
double fuga[9][10][11];
```

[問 4] 2点

```
bar[1][2]=foo;  
foo=bar[12][34];
```

## 2 文字列

[問 1] 6点

```
char hoge;  
hoge='A';  
printf("%c",hoge);
```

[問 2] 6点

```
char hoge[10];  
strcpy(hoge,"Akita");  
printf("%s",hoge);
```

[問 3] 6点

```
char hoge[10];  
strcpy(hoge,"秋田");  
printf("%s",hoge);
```

## 3 位取り記数法と 2 進数, 10 進数, 16 進数

[問 1] 各 1 点

[ア] 1111

[イ] 10000

[ウ] 10001

[エ] d

[オ] e

[カ] f

[キ] 10

[ク] 11

[問 2] 4 点

$$\begin{aligned}(10111101)_2 &= (1 \times 10^{11} + 0 \times 10^{10} + 1 \times 10^{101} + 1 \times 10^{100} + 1 \times 10^{11} + 1 \times 10^{10} + 0 \times 10^1 + 1 \times 10^0)_2 \\ &= (1 \times 2^7 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^0)_{10} \\ &= (128 + 32 + 16 + 8 + 4 + 1)_{10} \\ &= (189)_{10}\end{aligned}$$

16 進数に直すためには、二進数を下位から 4 ビットずつ区切る。この 4 ビットが 16 進数の各桁に対応する。問題の二進数の  
のもっと下位の 4 ビットを 10 真数に直すと 13 である。これは 16 進数の d に対応する。つぎの 4 ビットは 10 真数で 11 で、  
16 進数 b である。したがって、

$$(10111101)_2 = (bd)_{16}$$

となる。

[問 3] 4点

2進数への変換は以下のようにする.

$$\begin{array}{r}
2 \overline{) 261} \text{ --- } 1 \\
2 \overline{) 130} \text{ --- } 0 \\
2 \overline{) 65} \text{ --- } 1 \\
2 \overline{) 32} \text{ --- } 0 \\
2 \overline{) 16} \text{ --- } 0 \\
2 \overline{) 8} \text{ --- } 0 \\
2 \overline{) 4} \text{ --- } 0 \\
2 \overline{) 2} \text{ --- } 0 \\
1
\end{array}$$

したがって,

$$(261)_{10} = (100000101)_2 \quad (1)$$

である.

16進数へは2進数の下位4桁ずつ区切ることにより容易に変換できる.

$$(100000101)_2 = (105)_{16} \quad (2)$$

[問 4] 4点

16進数の一桁は2進数の4桁なので,16進数の各桁を2進数に変換すればよい.16進数の4は10進数でも4,二進数では0100である.16進数のbは10進数で11なので,二進数で1011である.したがって,2進数へ変換すると次のようになる.

さきほどの16進数の各桁の10進数の値を利用すると,10進数の値は次のようにすれば得られる.

$$(b4)_{16} = (11 \times 16 + 4)_{10} = (180)_{10} \quad (4)$$

$$(b4)_{16} = (10110100)_2 \quad (3)$$

## 4 ポインター

### 4.1 メモリーとデータ

[問 1] 2点  
8

[問 2] 2点  
32

[問 3] 2点  
8

[問 4] 2点  
4

[問 5] 3点

文字型:1バイト      整数型:4バイト      倍精度実数型:8バイト

### 4.2 ポインターと演算子

[問 1] 3点

メモリのアドレスを格納する変数のこと.(正確には,オブジェクトを示すものをポインターと言う.)

[問 2] 2点

```
int *pi;
double *px;
```

[問 3] 2点

```
pi=&iv;
```

[問 4] 2点

```
j=*pi;
```

[問 5] 各1点

[ア] 11223344

[イ] bfff6b0

[ウ] 11223344

[問 6] 4点

メモリの内容は以下のようになっている。

アドレス	記憶内容
ffff6b0	44
ffff6b1	33
ffff6b2	22
ffff6b3	11
ffff6b4	b0
ffff6b5	f6
ffff6b6	ff
ffff6b7	bf

### 4.3 ポインタの演算と関数の引数

[問 1] 2点

和と差

[問 2] 3点

関数に値そのものを渡す方法。

[問 3] 3点

関数に値のあるアドレスを渡す方法。

[問 4] 各1点

[ア] 0xbf0e98d

[イ] 0xbf0e8fc

[ウ] 0x804841e

## 5 プログラム作成

### 5.1 ポインタの応用

10点

```
#include <stdio.h>

void operation(int a, int b, int *w, int *s, int *m, int *d);

//===== メイン関数 =====
int main(void)
{
    int wa, sa, seki, sho;

    operation(100,26,&wa, &sa, &seki, &sho);

    printf("wa=%d\tsa=%d\tseki=%d\tsho=%d\n",wa ,sa ,seki ,sho);

    return 0;
}

//===== 四則演算の関数 =====
void operation(int a, int b, int *w, int *s, int *m, int *d)
{
    *w = a+b;
    *s = a-b;
    *m = a*b;
    *d = a/b;
}
```

## 5.2 配列の応用

5点

```
#include <stdio.h>

int main(void)
{
    FILE *in_file;
    int data[10000][4];
    int i, sum0=0, sum1=0, sum2=0, sum3=0;
    const int N=10000;

    in_file = fopen("/tmp/int_data.txt", "r");

    for(i=0; i<N; i++){
        fscanf(in_file,"%d%d%d%d",
            &data[i][0],&data[i][1],&data[i][2],&data[i][3]);
    }

    fclose(in_file);

    for(i=0; i<N; i++){
        sum0=data[i][0];
        sum1=data[i][1];
        sum2=data[i][2];
        sum3=data[i][3];
    }

    printf("all sum = %d\n", sum0+sum1+sum2+sum3);
    printf("average row_1=%f\trow_2=%f\trow_3=%f\trow_4=%f\n",
        (double)sum0/N, (double)sum1/N, (double)sum2/N, (double)sum3/N);

    return 0;
}
```

[注意] わざわざ、データ読み込みの部分と合計の計算部分を分ける必要はない。ここではわかりやすくするために分けただけである。