

学籍番号 _____ 氏名 _____

1. 構造体 (計 18 点)

1.1 データ構造 (計 10 点)

[問 1] (1 点)

```
double seisu;
```

[問 2] (1 点)

```
int hairetu[100];
```

[問 3] (2 点)

```
struct kozotai{
    int mi;
    double mx;
};
```

[問 4] (2 点)

変数に 1 つの値しか代入できないデータ構造を単純型と言う。

[問 5] (2 点)

順序付けられた複数の同じ型のデータを取り扱うことができる。配列名と添え字によりデータを指定できる。

[問 6] (2 点)

複数の異なった型のデータを一つにまとめることが出来るデータ構造である。

1.2 構造体基本 (計 8 点)

[問 1] (2 点)

```
struct student{
    char name[64];
    int english;
    int mathematics;
    double weight;
};
```

[問 2] (2 点)

```
struct student akita_kosen[50];
```

[問 3] (2 点)

```
akita_kosen[29].mathematics = 65;
```

[問 4] (2 点)

```
m = akita_kosen[29].mathematics;
```

2. 基数の変換 (計 22 点)

[問 1] (各 1 点)

[ア] 1110

[イ] 1111

[ウ] 10000

[エ] a

[オ] b

[カ] c

[キ] d

[ク] e

[ケ] f

[コ] 10

[問 2] (4 点)

$$\begin{aligned}(10110101)_2 &= (1 \times 2^7 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^2 + 1 \times 2^0)_{10} \\ &= (128 + 32 + 16 + 4 + 1)_{10} \\ &= (181)_{10}\end{aligned}$$

2 進数の 4 桁が 16 進数の 1 桁に対応するので、2 進数を 4 桁ずつに区切り、それぞれを 16 進数に変換する。

$$\begin{aligned}(1011)_2 &= (8 + 2 + 1)_{10} = (b)_{16} \\ (0101)_2 &= (4 + 1)_{10} = (5)_{16}\end{aligned}$$

したがって、 $(10110101)_2 = (b5)_{16}$ となる。

[問 3] (4 点)

$$(267)_{10} = (256+8+2+1)_{10} \\ = (100001011)_2$$

別解 (筆算をつかう)

$$\begin{array}{r} 2 \overline{) 267} \quad - 1 \\ 2 \overline{) 133} \quad - 1 \\ 2 \overline{) 66} \quad - 0 \\ 2 \overline{) 33} \quad - 1 \\ 2 \overline{) 16} \quad - 0 \\ 2 \overline{) 8} \quad - 0 \\ 2 \overline{) 4} \quad - 0 \\ 2 \overline{) 2} \quad - 0 \\ 1 \end{array}$$

2 進数の 4 桁が 16 進数の 1 桁に対応するので、2 進数を 4 桁ずつに区切り、それぞれを 16 進数に変換する。

$$\begin{array}{lll} (1)_2 & = (1)_{10} & = (1)_{16} \\ (0000)_2 & = (0)_{10} & = (0)_{16} \\ (1011)_2 & = (8+2+1)_{10} & = (b)_{16} \end{array}$$

したがって、 $(267)_{10} = (10001011)_2 = (10b)_{16}$ となる。

[問 4] (4 点)

16 進数の 1 桁は、2 進数の 4 桁に変換される。

$$\begin{array}{llll} (c)_{16} & = (12)_{10} & = (8+4)_{10} & = (1100)_2 \\ (8)_{16} & = (8)_{10} & = (8)_{10} & = (1000)_2 \end{array}$$

$$\begin{array}{l} (c8)_{16} = (12 \times 16^1 + 8 \times 16^0)_{10} \\ = (192 + 8)_{10} \\ = (200)_{10} \end{array}$$

したがって、 $(c8)_{16} = (11001000)_2$ となる。

3. コンピューターの構造とメモリー (計 26 点)

3.1 基本 (計 14 点)

[問 1] (各 1 点)

[ア] 記憶装置 [イ] 演算装置 [ウ] 出力装置 [エ] 制御装置 [オ] 入力装置

[問 2] (各 3 点)

主記憶装置の説明

プログラム (命令とデータ) を格納する装置で、直接 CPU と命令とデータの受け渡しを行う。

補助記憶装置の説明

大容量で主記憶装置に入りきらないデータや半永久的に残したいデータを格納する装置である。

[問 3] (1 点)

制御装置と演算装置

[問 4] (1 点)

ディスプレイ
プリンター

[問 5] (1 点)

キーボード
マウス

3.2 メモリーとデータの関係 (計 12 点)

[問 1] (1 点)

64 ビット
16 桁

[問 2] (1 点)

8 ビット

[問 3] (1 点)

32 ビット

[問 4] (1 点)

8 ビット

[問 5] (5 点)

メモリーはデータを記憶する装置である。そこでは、データは 8 ビット毎にまとめられて、記憶される。この 8 ビット毎のデータの記憶場所をアドレスといい、整数の $(00000000)_{16} \sim (ffffff)_{16}$ まで番号が振ってあり、これを指定することによりデータの格納と呼び出しができる。

[問 6] (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. ポインター (計 23 点)

[問 1] (3 点)

アドレスを格納する変数である。

[問 2] (2 点)

```
int *pint;
double *pdouble;
```

[問 3] (2 点)

```
ip=&iv;
```

[問 4] (2 点)

```
j = *ip;
```

[問 5] (3 点)

- ア `ffddaa88`
- イ `bffff690`
- ウ `ffddaa88`

[問 6] (6 点)

アドレス	記憶内容
<code>bffff690</code>	<code>88</code>
<code>bffff691</code>	<code>aa</code>
<code>bffff692</code>	<code>dd</code>
<code>bffff693</code>	<code>ff</code>
<code>bffff694</code>	<code>90</code>
<code>bffff695</code>	<code>f6</code>
<code>bffff696</code>	<code>ff</code>
<code>bffff697</code>	<code>bf</code>

[問 7] (5 点)

引き続く数が、16 進数ということを表している。

5. プログラム作成 (11 点)

```
#include <stdio.h>

struct seiseki{
    int kokugo;
    int sansu;
};

int main(void){

    struct seiseki shimada, yamada;

    shimada.kokugo=49;
    shimada.sansu=84;

    yamada.kokugo=65;
    yamada.sansu=75;

    printf("heikin shimada = %d\n", (shimada.kokugo+shimada.sansu)/2);
    printf("heikin yamada  = %d\n", (yamada.kokugo+yamada.sansu)/2);

    printf("heikin sansu   = %d\n", (shimada.sansu+yamada.sansu)/2);
    printf("heikin kokugo  = %d\n", (shimada.kokugo+yamada.kokugo)/2);

    return 0;
}
```