

<演習問題> ポインタ(1)

演習用マシンを使用する場合について答えよ。演習問題の回答はこの用紙に記入してレポートに挿入する。なお、本演習問題では、番地を10進数で表示する。

【演習問題 1】 以下の問に答えよ。

- (1) 以下のプログラムの実行結果を記述しなさい。ただし、変数 a は 1000 番地に、b は 2000 番地に格納されているものとする。

```
#include <stdio.h>

int main(void)
{
    int a = 100;
    float b = 18.25;
    printf("a の値は%d, アドレスは%p です. \n", a, &a);
    printf("b の値は%f, アドレスは%p です. \n", b, &b);
    return 0;
}
```

[実行結果]

a の値は_____, アドレスは_____です.
b の値は_____, アドレスは_____です.

- (2) 以下のプログラムの下線部(1)から(12)を埋めなさい。

```
#include <stdio.h>

int main(void)
{
    int a = 100;
    float b = 18.25;
    (1) _____ (変数 a のアドレスを格納するポインタ変数 pa の宣言文)
    (2) _____ (変数 b のアドレスを格納するポインタ変数 pb の宣言文)
    (3) _____ (変数 a のアドレスを変数 pa に代入する文)
    (4) _____ (変数 b のアドレスを変数 pb に代入する文)
    printf("a の値は (5) _____ です. \n, (6) _____); (変数 a の値を pa を使って表示する文)
    printf("b の値は (7) _____ です. \n, (8) _____); (変数 b の値を pb を使って表示する文)
    printf("a のアドレスは (9) _____ です. \n, (10) _____); (変数 a のアドレスを pa を使って表示する文)
    printf("b のアドレスは (11) _____ です. \n, (12) _____); (変数 b のアドレスを pb を使って表示する文)
    return 0;
}
```

(3) 以下のプログラムの実行結果を記述しなさい。ただし a[0]は 3000 番地に格納されているものとする。

```
#include <stdio.h>
int main( void )
{
    int a[4] = { 100, 200, 300, 400 };

    printf( "&a[0]=%p, &a[0]=%p, a=%p, *(&a[0])=%d, *(a)=%d\n",
           &a[0], &a[0], a, *(&a[0]), *(a) );

    printf( "&a[1]=%p, &a[0]+1=%p, a+1=%p, *(&a[0]+1)=%d, *(a+1)=%d\n",
           &a[1], &a[0]+1, a+1, *(&a[0]+1), *(a+1) );

    printf( "&a[2]=%p, &a[0]+2=%p, a+2=%p, *(&a[0]+2)=%d, *(a+2)=%d\n",
           &a[2], &a[0]+2, a+2, *(&a[0]+2), *(a+2) );

    printf( "&a[3]=%p, &a[0]+3=%p, a+3=%p, *(&a[0]+3)=%d, *(a+3)=%d\n",
           &a[3], &a[0]+3, a+3, *(&a[0]+3), *(a+3) );

    return 0;
}
```

[実行結果]

```
&a[0]=_____, &a[0]=_____, a=_____, *(&a[0])=_____, *(a)=_____
&a[1]=_____, &a[0]+1=_____, a+1=_____, *(&a[0]+1)=_____, *(a+1)=_____
&a[2]=_____, &a[0]+2=_____, a+2=_____, *(&a[0]+2)=_____, *(a+2)=_____
&a[3]=_____, &a[0]+3=_____, a+3=_____, *(&a[0]+3)=_____, *(a+3)=_____
```

【演習問題 2】

変数 a	1000 番地		変数 c	2000 番地	
変数 pa	xxx 番地				
			変数 pc	yyy 番地	

int a 宣言により，メモリ上に整数型変数 a の値を格納するための領域が 1000 番地から(1)_____バイト用意されるものとする．また，double c 宣言により，メモリ上に実数型変数 c の値を格納するための領域が 2000 番地から(2)_____バイト用意されるものとする．

「変数 a に値 5 を格納する」と「1000 番地に値 5 を格納する」の 2 つの文は同じことを表す. 「変数 a から値を取り出す」と「1000 番地から格納されている値を取り出す」の 2 つの文は同じことを表す.

変数 a の番地を記憶するための変数 pa ((3) _____ 変数と呼ぶ) を用意する。
変数 pa の宣言文の記法は？ (4) _____ ；

変数 `c` の番地を記憶するための変数 `pc` を用意する. 変数 `pc` の宣言文の記法は?
(5) _____ ;

変数 a の番地を変数 pa に格納することを示す文は？
(6)_____；

変数 c の番地を変数 pc に格納することを示す文は？
(7)_____；

pa に格納された番地に格納された値を整数型変数 b に格納することを示す文は？
(8)_____；

pa に格納された番地に値 100 を格納することを示す文は？
(9)_____；

変数 pa の領域の大きさは(10)_____バイト. 変数 pc の領域の大きさは(11)_____バイト.

pa++; および pc++; を実行した結果，変数 pa および pc に格納されている値はいくつか？
pa : (12)_____番地 pc : (13)_____番地

b=a+10; という算術式を，変数 a, b を使わず，変数 a の番地を格納してある変数 pa と変数 b の番地を格納してある変数 pb を使用して書き直すとどうなるか？
(14)_____；

【演習問題 3】

以下のプログラムを実行したとき、表示される値を余白に記述せよ。ただし、配列 `idata` は 1000 番地から格納されているとする。また、表示結果 8-10 の箇所、正しく計算された値のみを表示させるには、`max` はいくらくらに設定すればよいか。

```
#include<stdio.h>
main()
{
    int iv, idata[]={2,4,6,8,10,12}, *ip, i, max ;

    iv=idata[0];
    printf("*1* iv の値=%d\n",iv);           表示結果 1 :      *1* iv の値 = 2
    ip=&idata[0];
    printf("*2* ip の値=%p\n",ip);           表示結果 2 :
    ip=&idata[1];
    printf("*3* ip の値=%p\n",ip);           表示結果 3 :
    ip=&idata[2];
    printf("*4* ip の値=%p\n",ip);           表示結果 4 :

    iv=*ip;
    printf("*5* iv の値=%d\n",iv);           表示結果 5 :
    iv=*ip+3;
    printf("*6* iv の値=%d\n",iv);           表示結果 6 :
    iv=*(ip+3);
    printf("*7* iv の値=%d\n",iv);           表示結果 7 :

    max = ? ;                               max の値 :

    *ip=0;
    for(i=0; i<max; i++){
        printf("*8* *(ip+%d)= %d\n",i,*(ip+i)); 表示結果 8 :
    }

    *(ip+1)=10;
    for(i=0; i<max; i++){
        printf("*9* *(ip+%d)= %d\n",i,*(ip+i)); 表示結果 9 :
    }

    *(ip+3)=20;
    for(i=0; i<max; i++){
        printf("*10* *(ip+%d)=%d\n",i,*(ip+i)); 表示結果 10 :
    }

    ip=&idata[0];
    printf("*11* ip の値=%p\n",ip);           表示結果 11 :
    ip=ip+2;
    printf("*12* ip の値=%p\n",ip);           表示結果 12 :

    iv=*ip+3;
    printf("*13* iv の値=%d\n",iv);           表示結果 13 :
    iv=*(ip+3);
    printf("*14* iv の値=%d\n",iv);           表示結果 14 :

    ++ip;
    printf("*15* ip の値=%p\n",ip);           表示結果 15 :
}
```

【演習問題 4】

下記のプログラムの文①～⑤を実行した後のメモリ内容の値を下のメモリ図に記入せよ.

```
#include<stdio.h>
void swap(int *x, int *y );
int main()
{
    int a, b ;
    a=12; b=34;           ①
    swap(&a, &b);          ②
}

void swap( int *x, int *y )
{
    int wk=0 ;

    wk=*x;   ③
    *x=*y;   ④
    *y=wk;   ⑤
}
```

(1) ①を実行した後の main 関数のメモリの内容

		メモリ
変数名	アドレス	内容
b	1000	
a	1004	

(2) ②の swap(&a, &b) 関数が呼び出されたときの swap 関数のメモリの内容

		メモリ
変数名	アドレス	内容
wk	2000	
y	2004	
x	2008	

(3) ③④⑤を実行したときの main 関数のメモリの内容と swap 関数のメモリの内容

		メモリ
変数名	アドレス	内容
b	1000	
a	1004	

		メモリ
変数名	アドレス	内容
wk	2000	
y	2004	
x	2008	

【演習問題 5】

以下のプログラムは、指定された人数分の試験の点数（素点）を読み込み、43点の出席点を加算した後、全員の点の合計を計算するものである。①～⑤の番号がある行は誤りがある行である。横の例にならって間違いを訂正せよ。また、訂正したプログラムを作成し、実行させなさい。

```
#include<stdio.h>
#define MAXNUM 10
int main(void)
{
    int nin; /* 人数 */
    int data[MAXNUM], *pdata; /* データの格納領域 */
    int goukei; /* 合計点数 */
    int i; /* 繰返し制御変数 */

    scanf("%d", &nin); /* 人数読み込み */
    for(i=0; i<nin; i++) /* 素点読み込み */
    {
        scanf("%d", &data[i]);
    }

    pdata=*data; /* ポインタ変数初期化 */

    for(i=0; i<nin; i++)
    {
        *pdata++=(data[i]+43); /* ポインタ変数を用いた */
        /* 出席点追加 */
    }

    pdata=&data[0]; /* ポインタ変数初期化 */
    goukei=pdata; /* ポインタ変数を用いた */
    /* 合計計算の初期値設定 */

    for(i=1; i<=nin; i++) /* ④ */
    {
        goukei += *pdata; /* ⑤ */
        /* ポインタ変数を用いた */
        /* 合計点計算 */
    }
    printf("合計=%d¥n", goukei);
    return(0);
}
```

①～⑤番の行のプログラムの修正

① scanf("%d", &nin);

②

③

④

⑤