

課題 3

Kadai3-1

文字型配列 `string` に、“ABCDEFGG” という文字列が格納されている。文字型配列 `rstring` にこの文字列を逆順に格納し、表示しなさい。ただし、文字を逆順に格納する処理についてはポインタ変数を利用すること。また、以下の図はヒントである。これを参考にしてプログラムの空欄を埋めること。

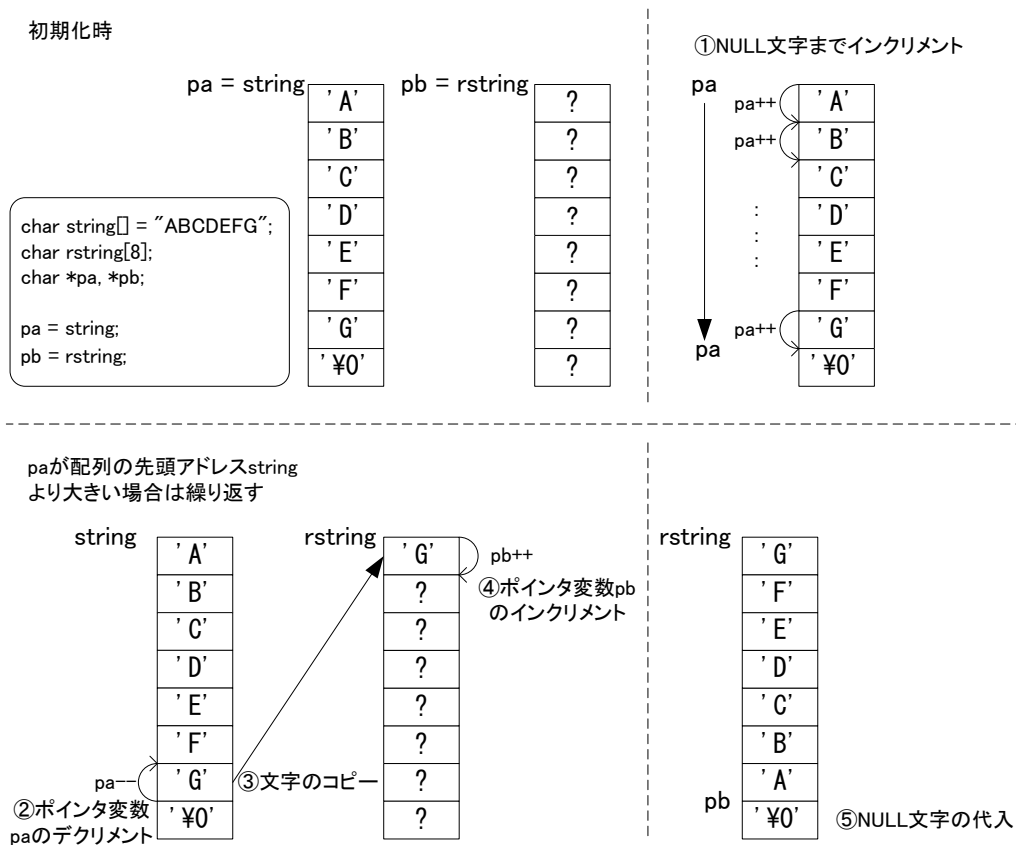


図 1 Kadai3-1 のヒント

```

Kadai3-1
#include <stdio.h>

int main( void )
{
    char string[] = "ABCDEFGG";
    char rstring[8];
    char *pa, *pb;

    pa = string;
    pb = rstring;

    while ( 【 ① 】 ) // ①NULL文字までポインタ変数の値をインクリメント
    {
        pa++;
    }

    // while文の終了時には*paは'\0'となっている
    
```

```

while (string < pa) // 配列の先頭まで逆順でコピーしていく
{
    【 ② 】 ; // ②paをデクリメント
    【 ③ 】 ; // ③文字のコピー
    【 ④ 】 ; // ④pbをインクリメント
}

【 ⑤ 】 ; // ⑤最後にNULL文字を代入する

printf("string :%s\n", string);
printf("rstring :%s\n", rstring);

return 0;
}

```

実行結果

```

string :ABCDEFGG
rstring :GFEDCBA

```

Kadai3-2

要素数 10 の整数型配列 data を用意して下図のように初期化しなさい。この配列を順に調べ、偶数の値のみ、要素数 10 の整数型配列 result に代入すること。その際、何個格納したかをカウントしておく。結果として偶数の個数とその数値を全て表示しなさい。

ただし、配列の要素へのアクセスはポインタを用いて行うこと。配列 data、result の先頭アドレスをそれぞれポインタ (int *pa, *pb;など) に代入し、ポインタの値をインクリメント (pa++や pb++) することでプログラムを作ること。

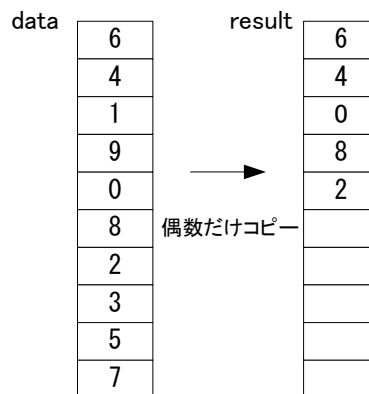


図 2 Kadai3-2 のヒント

実行結果

```

偶数の数 : 5
6 4 0 8 2

```

Kadai3-3

要素数 10 の整数型配列 `sum` に対して、以下の図のようにして、その配列の添え字までの総和を格納し、表示しなさい。

ただし、配列の要素へのアクセスはポインタ (`int *p;` など) を用いて行うこと。配列 `sum` の先頭アドレスをそれぞれポインタ (`int *p;` など) に代入し、ポインタの値をインクリメント (`p++`) することでプログラムを作成すること。

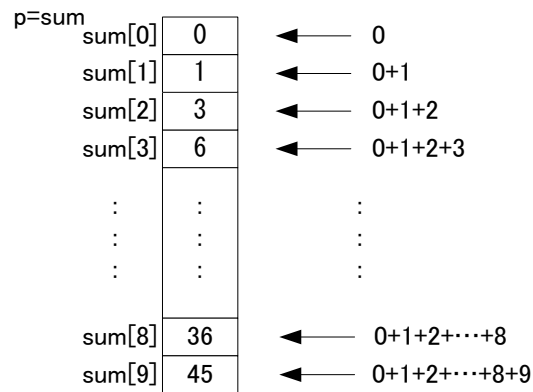


図3 Kadai3-3 のヒント (1)

以下の図のように、`*p` に一つ前の要素の `*(p-1)` に添字の値を加えたものを格納するとシンプルなプログラムになる。ただし、先頭要素は `*(p-1)` を加えるわけにはいかない (配列の領域外に指し示してしまう) ので、先頭要素には先に 0 を代入し、それから `p++` して 2 番目の要素から処理を行う。

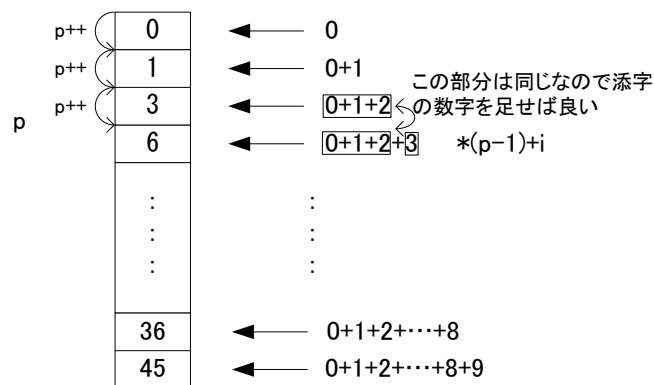


図4 Kadai3-3 のヒント (2)

実行結果

0 1 3 6 10 15 21 28 36 45