

1 リスト処理トレース

```
struct List {
    int data ;
    struct List* next ;
} ;
struct List* primes = NULL ;
void main() {
    int i ;
    struct List* p ;
    for( i = 2 ; i <= 10 ; i++ ) {
        for( p = primes ; p != NULL ; p = p->next ) {
            if ( i % p->data == 0 )
                break ;
        }
        if ( p == NULL )
            primes = cons( i , primes ) ;
    }
    /* (A) */
    for( p = primes ; p != NULL ; p = p->next )
        printf( "%d " , p->data ) ;
}
```

```
struct List* cons( int x , struct List* n ) {
    struct List* ans =
        (struct List*)malloc( sizeof( struct List ) ) ;
    if ( ans != NULL ) {
        ans->data = x ;
        ans->next = n ;
    }
    return ans ;
}
```

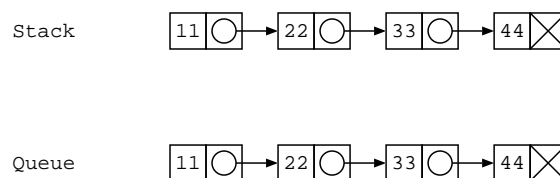
設問 1 このプログラムの
実行結果を答えよ。(3x4)

設問 2 (A) の時点で生成
されたデータ構造の
イメージ図を示せ。(2x4)

2 リスト処理 Stack, Queue

以下の図を用いて、リスト構造を用いたスタック、待ち行列を実装する際の、処理について説明せよ。(5x4)

ただし、(1) データを管理するための変数やその型が明記されていること。(2) 説明では、スタックへのデータ追加 push(), 取り出し pop(), 待ち行列への追加 put(), 取り出し get() などを図や説明を交えて示すこと。(3) ポインタの繋ぎ替えなどが図で説明されていれば、具体的なプログラムを示さなくても良い。

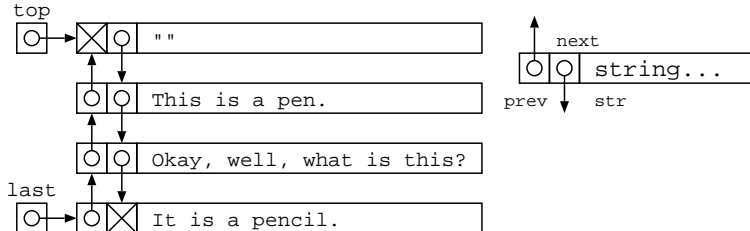


3 双方向リスト

```

struct StrDeque {
    char str[ 80 ] ;
    struct StrDeque* prev ;
    struct StrDeque* next ;
} ;

struct StrDeque* top ;
struct StrDeque* last ;
    
```



```

struct StrDeque* sdcons( struct StrDeque* p , struct StrDeque* n ,
                        char s[] ) {
    struct StrDeque* ans =
        (struct StrDeque*)malloc( sizeof( struct StrDeque ) ) ;
    if ( ans != NULL ) {
        ans->prev = p ;
        ans->next = n ;
        strcpy( ans->str , s ) ;
    }
    return ans ;
}

void append( char s[] ) {
    last->next =
        sdcons( last , NULL , s ) ;
    last = last->next ;
}

void find( char key[] ) {
}

void main() {
    struct StrDeque* p ;
    top = last = sdcons( NULL , NULL , "" ) ;
    append( "This is a pen." ) ;
    append( "Okay, well, what is this?" ) ;
    append( "It is a pencil." ) ;
    find( "pen" ) ;
}
    
```

// これ以降を作成せよ。
// この例であれば、"line=1,column=10","line=3,column=8"を表示する処理を記述せよ。

4 型と説明

前設問のプログラムにおいて、以下の設問に答えよ。

- (A)~(D) の型を答えよ。

(A) _____, (B) _____, (C) _____, (D) _____

- データ構造の先頭に、空文字列が入ったデータが作られているが、このようなダミーデータの一般的な名称と、その目的を説明せよ。

5 リストによる集合処理

集合 $A=\{1,2,3,4,5\}$, $B=\{2,4,6,8\}$ が与えられたとき、差集合 $A-B$ を求める関数を、リスト構造を用いて記述したい。関数 `diff(a,b)` を作成せよ。

```
// cons() は設問1のプログラムを利用
struct List* a = cons(1, cons(2, cons(3, cons(4, cons(5, NULL)))));
struct List* b = cons(2, cons(4, cons(6, cons(8, NULL)))));

struct List* diff( struct List* a , struct List* b ) {
    // {1,3,5}が求まるような処理であること
}
```

6 応用問題

以下に示すような、星座名と星座の期間の最終の月日のリスト `atop` と、家族の名前と誕生日のリスト `mtop` が大域変数で与えられるとする。

図に示すデータ構造にふさわしい宣言を示し、家族の名前と星座の一覧を表示するプログラムを作成せよ。

ただし、星座名リストや家族リストは、あらかじめ宣言されているものとし、データを初期化する処理を記述する必要はない。

