

プログラミング基礎演習

レポート課題1

～ヒント～

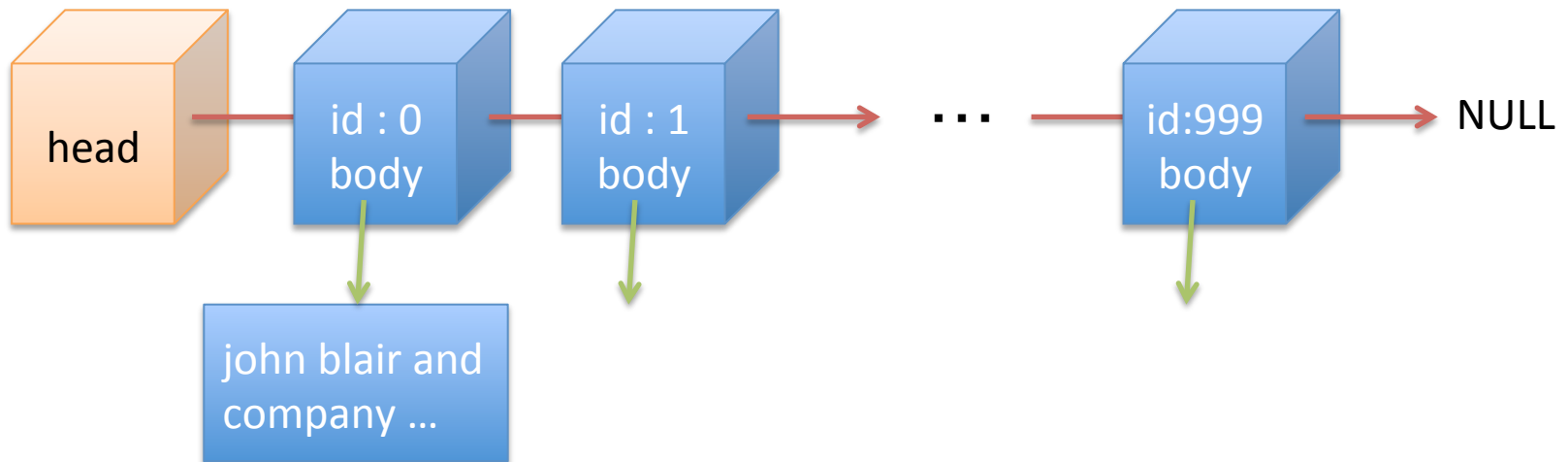
2013/1/2

岩成達哉

まずはIDを振ったデータの集まりを作る

0	john blair and company ...
1	industry sources put ...
2	john blair was acquired ...
:	
999	agreement that repackages ...

どんなデータ構造でもいいが、ヒントで挙げられている構造体で線形リストをつくると



のような形

どうやってつくる？

1. ファイルから一行単位でbodyをつくる

ファイルを開いて読み出していく

いくつか方法があるが、例として、

- 方法1: `fgets()`という関数で1行ずつ読み込む
- 方法2: `fgetc()`で、1文字ずつ見ていって、`'\n'`で区切るなどがある

2. bodyができれば線形リストに入れていく

これは今までの課題と同じ

idはインクリメント(+1)をしていって順次代入すればいい

次は転置索引を作る

john	{0, 2, ...}
blair	
and	
:	
credit	

これまたどんなデータ構造でもいい
伏見がメールでヒントとしてリストアップしてくれた関数群のようなときは、
(説明の簡単化のために線形リストで作ると)

`struct indices`

```
{  
    char *word; // johnやblairなど  
    int *index; // wordが出現する文のID{0, 2, ...}などに相当  
    struct indices *next;  
};
```

みたいな感じ(図はさっきとほとんど同じ)

どうやってつくる？

とりあえず単語のリストを作ることを考えよう

1. 文章をもう一度読みだす

読み取るのは、先程作成した線形リストのbody(か、もしくは指定がないので、ファイルをもう一度開いて処理するのもいいのかな)

2. 単語の切り出しを行う

bodyは線形リストを最初から最後までループで回してみていく
(ファイルからなら今まで通り)

- 方法1: 1文字ずつ読み出す
- 方法2: strtok()を使う

など、好きな方法でどうぞ

※strtok()を使う場合は、最初に入れた文字列が変更されてしまうので、bodyをそのまま入れたらだめ

適当な変数を用意して、strcpy()でコピーを作ってそいつを渡そう

実はちょっと工夫すればいい

3. 単語を切り出したら、線形リストに追加

同じ単語が追加されていないかチェックしながら、線形リストに追加していこう
このとき、追加の関数に渡すのは、単語だけじゃなくて今見ている行数
(bodyかファイルの行数)も

- 新しい単語なら
 - 構造体の領域を確保して、wordを保存
 - さらに、indexの領域を確保して、行数を入れておこう
- すでに保存されている単語なら
 - indexの領域を拡張して、今の行数も保存しておこう
realloc()でもいいし、新しくmalloc()してもいい

こんなんでも転置索引できちゃう

あとは検索！

1. ユーザから入力を受け付ける

ex) john blair said

一文字ずつ読み込みでも, `fgets()`などを使ってもいい

`scanf`だと空白を読まないなので工夫が必要(wikipedia見てみよう)

2. 取得した文を解析

といっても空白で区切れればいいだけ

3. 転置索引と比較

最初の単語を読みだした時に, その単語をキーとして転置索引を引く

そして, `index`を適当な変数(`int *result`)で受け取ろう

次の単語からは, 転置索引を引いて結果を返すたびに, この変数とその結果に共通して含まれるものだけを残すようにすればいい

共通したものを残す

4. 積集合をつくる

indexの内容がA:{0, 1, 3}とB:{1, 3, 7, 9}のときは, {1, 3}ができるようにする
ループで回してチェックする

- A:0で固定してBの要素をすべて見ていく → ないので残さない
- A:1で固定してBの要素をすべて見ていく → あるので残す
- 以下同様

残す方法は,

- 新しく変数(int *型)を用意して領域を確保しながら, 代入していく
- Aの内容を変更していく

などいくつか方法がある

5. 入力された文を全て見たら残った結果(result)を使って, 行を表示する

一番最初に作った線形リストのIDと比較しながら表示すればいい

まとめ

説明のしやすいように線形リスト押しですが、別になんでもいいです

また、手順はヒントの順じゃなくていいはず

今まで習った内容の総復習になりそうですね

※ファイル名を指定する部分は省いてますが、忘れずに実装してください