

計算機システム演習 第二回レポート

17B13541

細木隆豊

1 説明

node の構造を `{int val; struct node *next; struct node *prev;};` で必要十分であるとして考えた。また、前後する 2 つの node の "prev" と "next" は互いの node のポインタが入るので、2 つの node のポインタを入れるとその 2 つをつなぐ関数 connect を実装した。

put は新たに "val" に引数の val をとる node を作り、元々の最後の node と tail の間にその node を付け加える (繋ぎ変える) ことを行った。失敗の判断は、新たに node を作れなかった、tail または元々最後の node のポインタが NULL の場合にしている。

get は最初の node の "val" を返し、head と二番目の node を繋ぐプログラムである。表示した値の node は解放するように実装した。データが存在しないとき (つまり head と tail しかない) 場合や、head, 最初の node のポインタが取れていない場合に -1 を返すようにした。

delete, display は head から tail の間の node に、val があるか確認したり、順に表示したりする関数である。delete では val が存在した場合、その node は解放している。

main 関数には、rand を用いても動作の確認をした。

2 実行結果

rand を用いない場合に実行した結果を以下に載せる。

This Queue is initial state.

Add numbers in the following.

0, 1, 2, 3, 4, 5, 6, 7, 8, 9,

This Queue includes the following.

0, 1, 2, 3, 4, 5, 6, 7, 8, 9,

Delete odds.

0, 2, 4, 6, 8,

11 isn't includes, so return -1

get first term 0

2, 4, 6, 8,

free all.

3 感想・質問

ポイントの扱いはだいぶ理解してきてはいるが、よく考えないと未だに合っているのかわからなくなるのでもう少し理解を深める為に自分なりに勉強しようと思う。