

## 第9回 プログラミング応用レポート

15302114 番 山下尚人

提出日：2018 年 1 月 9 日

(1)

ファイルにセットした値を読み込み，連結リストに値をセットする例

- ソースコード

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  struct CELL {
5      struct CELL *next;
6      int value;
7  };
8
9  void set_cell(struct CELL *data, int val){
10     struct CELL *temp;
11     temp = malloc(sizeof(struct CELL));
12
13     if(temp == NULL){
14         printf("Not enough memory\n");
15         exit(EXIT_FAILURE);
16     }
17
18     while(data->next!=NULL){
19         data = data->next;
20     }
21
22     temp->next = data->next;
23     temp->value = val;
24     data->next = temp;
25 }
26
27 void print_cell(struct CELL *data){
28     struct CELL *p;
29
30     for(p=data->next; p!=NULL; p=p->next){
31         printf("value=%d\n", p->value);
32     }
33 }
34
35 int main (void){
36     struct CELL data;
```

```

37     data.next = NULL;
38     data.value = 0;
39     int value;
40     FILE *fp;
41
42     if ((fp=fopen("data1.dat","r")) == NULL){
43         printf("File not found. ---data1.dat\n");
44         exit(EXIT_FAILURE);
45     }
46
47     while(fscanf(fp, "%d", &value) != EOF){
48         set_cell(&data,value);
49     }
50
51     print_cell(&data);
52     fclose(fp);
53
54     return EXIT_SUCCESS;
55 }

```

- 読み込まれたデータ

```

1     1
2     2
3     5
4     10
5     20
6     100

```

- 実行結果

```

1     value=1
2     value=2
3     value=5
4     value=10
5     value=20
6     value=100

```

- 考察

9～25 行目は set\_cell 関数を定義している。この関数は引数で指定したセルの一つ後ろに新しいセルを挿入する。引数は構造体 CELL 型のポイント型変数 data と、整数型の変数 val をとる。data は新しく後ろに挿入するセルの一つ前のポインタであり、val は新たに挿入するセルの値となる。

27～33 行目は print\_cell 関数を定義している。この関数は連結リストのセルの値を順番に標準出力に表示する。引数は構造体 CELL 型のポイント型変数 data をとる。data は表示するリストの先頭のセ

ルとなる。

35～55 行目は main 関数で、ファイルから読み込んだ値から set\_cell 関数で連結リストを作成。print\_cell 関数で作成したリストを標準出力に表示している。

36～38 行目は連結リストの先頭のダミーデータである、構造体 CELL 型の変数 data を定義している。

42～45 行目はファイルを読み込みモードで開き、ファイルが開けない場合は異常終了している。

47～49 行目は fscanf 関数によりファイルから値を読み込み、set\_cell 関数でセルを作成。それを while 文でファイルの終了まで行っている。

43 行目は print\_cell 関数により、47～49 行目で作成したリストの値を標準出力に表示している。

## (2)

ファイルに “10,9,8,7,6,4,3,2,” の値 (5 が抜けている) を順に書き込む

### ● ソースコード

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  struct CELL {
5      struct CELL *next;
6      int value;
7  };
8
9  void set_cell(struct CELL *data, int val){
10     struct CELL *temp;
11     temp = malloc(sizeof(struct CELL));
12
13     if(temp == NULL){
14         printf("Not enough memory\n");
15         exit(EXIT_FAILURE);
16     }
17
18     while(data->next!=NULL){
19         data = data->next;
20     }
21
22     temp->next = data->next;
23     temp->value = val;
24     data->next = temp;
25 }
26
27 void print_cell(struct CELL *data){
28     struct CELL *p;
29
30     for(p=data->next; p!=NULL; p=p->next){
31         printf("value=%d\n", p->value);
32     }
33 }
```

```

34
35 int main (void){
36     struct CELL data;
37     data.next = NULL;
38     data.value = 0;
39     int i;
40     FILE *fp;
41
42     if ((fp=fopen("data2.dat","w")) == NULL){
43         printf("File not found. ---data2.dat\n");
44         exit(EXIT_FAILURE);
45     }
46
47     for(i=10; i>0; i--){
48         if(i==5){
49             continue;
50         }
51         set_cell(&data,i);
52         fprintf(fp, "%d\n", i);
53     }
54
55     print_cell(&data);
56     fclose(fp);
57
58     return EXIT_SUCCESS;
59 }

```

#### ● 実行結果

```

1 value=10
2 value=9
3 value=8
4 value=7
5 value=6
6 value=4
7 value=3
8 value=2
9 value=1

```

#### ● 書き込まれたデータ

```

1 10
2 9
3 8
4 7
5 6
6 4
7 3
8 2
9 1

```

- 考察

課題 (1) の関数の定義をそのままに、42 行目以降の main 関数での処理を変更したプログラム。

“10,9,8,7,6,4,3,2,” という連結リストを作成し、ファイルに値を書き込んでいる。42～45 行目でファイルを書き込みモードで開いている。

47～53 行目で for 文により連番を作って set\_cell 関数でリストを作成し、fprintf 関数でファイルに書き込んでいる。48～50 行目の if 文による条件分岐で、5 だけを飛ばしている。

### (3)

連結リストにおいて “5” を 4 と 6 の間に挿入する

- ソースコード

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  struct CELL {
5      struct CELL *next;
6      int value;
7  };
8
9  void set_cell(struct CELL *data, int val){
10     struct CELL *temp;
11     temp = malloc(sizeof(struct CELL));
12
13     if(temp == NULL){
14         printf("Not enough memory\n");
15         exit(EXIT_FAILURE);
16     }
17
18     temp->next = data->next;
19     temp->value = val;
20     data->next = temp;
21 }
22
23 void print_cell(struct CELL *data){
24     struct CELL *p;
25
26     for(p=data->next; p!=NULL; p=p->next){
27         printf("value=%d\n", p->value);
28     }
29 }
30
31 int main (void){
32     struct CELL data, *p;
33     data.next = NULL;
34     data.value = 0;
35     int value;
36     FILE *fp;
```

```

37
38     if ((fp=fopen("data2.dat","r")) == NULL){
39         printf("File not found. ---data2.dat\n");
40         exit(EXIT_FAILURE);
41     }
42
43     while(fscanf(fp, "%d", &value) != EOF){
44         for(p=&data; p->next!=NULL; p=p->next);
45         set_cell(p,value);
46     }
47
48     printf("Before\n");
49     print_cell(&data);
50
51     printf("After\n");
52     for (p=data.next; p!=NULL; p=p->next){
53         if (p->value == 6){
54             set_cell(p,5);
55             break;
56         }
57     }
58     print_cell(&data);
59     fclose(fp);
60
61     return EXIT_SUCCESS;
62 }

```

#### ● 読み込まれたデータ

```

1    10
2     9
3     8
4     7
5     6
6     4
7     3
8     2
9     1

```

#### ● 実行結果

```

1    Before
2    value=10
3    value=9
4    value=8
5    value=7
6    value=6
7    value=4
8    value=3

```

```
9  value=2
10 value=1
11 After
12 value=10
13 value=9
14 value=8
15 value=7
16 value=6
17 value=5
18 value=4
19 value=3
20 value=2
21 value=1
```

- 考察

課題 (1) のプログラムから、set\_cell 関数の定義を少し変更した。

渡されたセル (連結リスト) の最後のセルの後ろに新しいセルを挿入するのではなく、渡されたセルの直後にセルを挿入する関数とした。

これにより 44 行目でファイルから値を読み込み連結リストを作る時、set\_cell 関数に渡すセルを連結リスト最後から一つ前のセルとしている。

46 行目まででファイルから値を読み込み、5 が抜けたデータを作成した。52～57 行目で 6 の後に 5 のセルを挿入している。