

モジュール設計仕様書	システム	家計簿.csv	家計簿入力補助・履歴検索	作成日	2020/5/18
補足	モジュール	Visual Studio C++ - CSV - Console	家計簿支出検索(家計簿入力補助)(C++-CSV-コンソール)	作成者	hosiymakaze

DFD:⑦-1 家計簿支出検索(家計簿入力補助) Visual Studio 2017 C++-CSV コンソールアプリ実装(CSVファイルを読み込む概念図)(map,vector,list版)

No.	日付	口座	大分類	中分類	小分類	項目	数量	金額	取引先	備考	周期
Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
96	2014/2/3(月)	SUICA	食費	外食	食事	トーストセット	1	399	ジョナサン	NULL	NULL
97	2014/2/3(月)	SUICA	食費	外食	食事	ヨーグルト	1	105	ジョナサン	NULL	NULL

```

63 //項目見出し(名称、位置)をMapに取得する
64 ItemIT_t next(strbuf, strbuf + strlen(strbuf), rxDel);
65 for (next = ItemIT_t(strbuf, strbuf + strlen(strbuf), rxDel, -1), i = 0; next != end; ++next, i++) {
66     ItemMap.insert(ItemMap_t::value_type(_strdup((char *)&next->str()[0]), i));
67 }

```

```
18 typedef std::map<char *, int> ItemMap_t;
19 ItemMap t ItemMap;
```

ItemMap	値
[comparator]	less
[allocator]	allocator
[0x000001be9fdaeca0 "No."]	0
[0x000001be9fdaf990 "小分類"]	5
[0x000001be9fdafb70 "大分類"]	3
[0x000001be9fdafbc0 "項目"]	6
[0x000001be9fdb0200 "数量"]	7
[0x000001be9fdb0250 "備考"]	10
[0x000001be9fdb02a0 "周期"]	11
[0x000001be9fdb0520 "金額"]	8
[0x000001be9fdb06b0 "口座"]	2
[0x000001be9fdb0700 "中分類"]	4
[0x000001be9fdb07f0 "日付"]	9
[0x000001be9fdb08e0 "取引先"]	1
[未加工データ]	{...}

```

72 //データ項目をVectorに取得する
73 ItemIT_t next(strbuf, strbuf + strlen(strbuf), rxDel);
74 for (next = ItemIT_t(strbuf, strbuf + strlen(strbuf), rxDel, -1), i=0; next != end; ++next, i++) {
75     rowVec.push_back(_strdup((char *)&next->str()[0]));
76 }
77 for (; i < ItemMap.size(); i++) {
78     rowVec.push_back(_strdup("")); //最後の項目値がNULLの分
79 }

```

```
22  vector<char*> rowVec;
```

rowVec	{ size = 12 }
[capacity]	13
[allocator]	allocator
[0]	0x00001be9fdb3b40 "86"
[1]	0x00001be9fdb0430 "2014/2/3(月)"
[2]	0x00001be9fdb0570 "SUICA"
[3]	0x00001be9fdb05c0 "食費"
[4]	0x00001be9fdb0610 "外食"
[5]	0x00001be9dfafa80 "食事"
[6]	0x00001be9dfadfa0 "トーストセット"
[7]	0x00001be9fdb3c80 "1"
[8]	0x00001be9fdb0660 "399 "
[9]	0x00001be9fdb0750 "ジョナサン"
[10]	0x00001be9fdb3740 ""
[11]	0x00001be9fdb3540 ""
[未加工ビュー]	{...}

```
13 const char *strDaiBnr = "大分類";
14 const char *strChuBnr = "中分類";
15 const char *strShoBnr = "小分類";
16 const char *strKmk = "項目";
```

```
83 //データ項目を出力対象項目(大分類、中分類、小分類、項目をtab区切りでつなぐ)のみに絞る
84 sprintf_s(strbuf, "%s\t%s\t%s\t%s", strValue(strDaiBnr), strValue(strChuBnr), strValue(strShoBnr), strValue(strKmk));
85 if(isNewValue(strbuf)) DataList.push_back( strdup(strbuf));
```

```
25 list<char *> DataList;
```

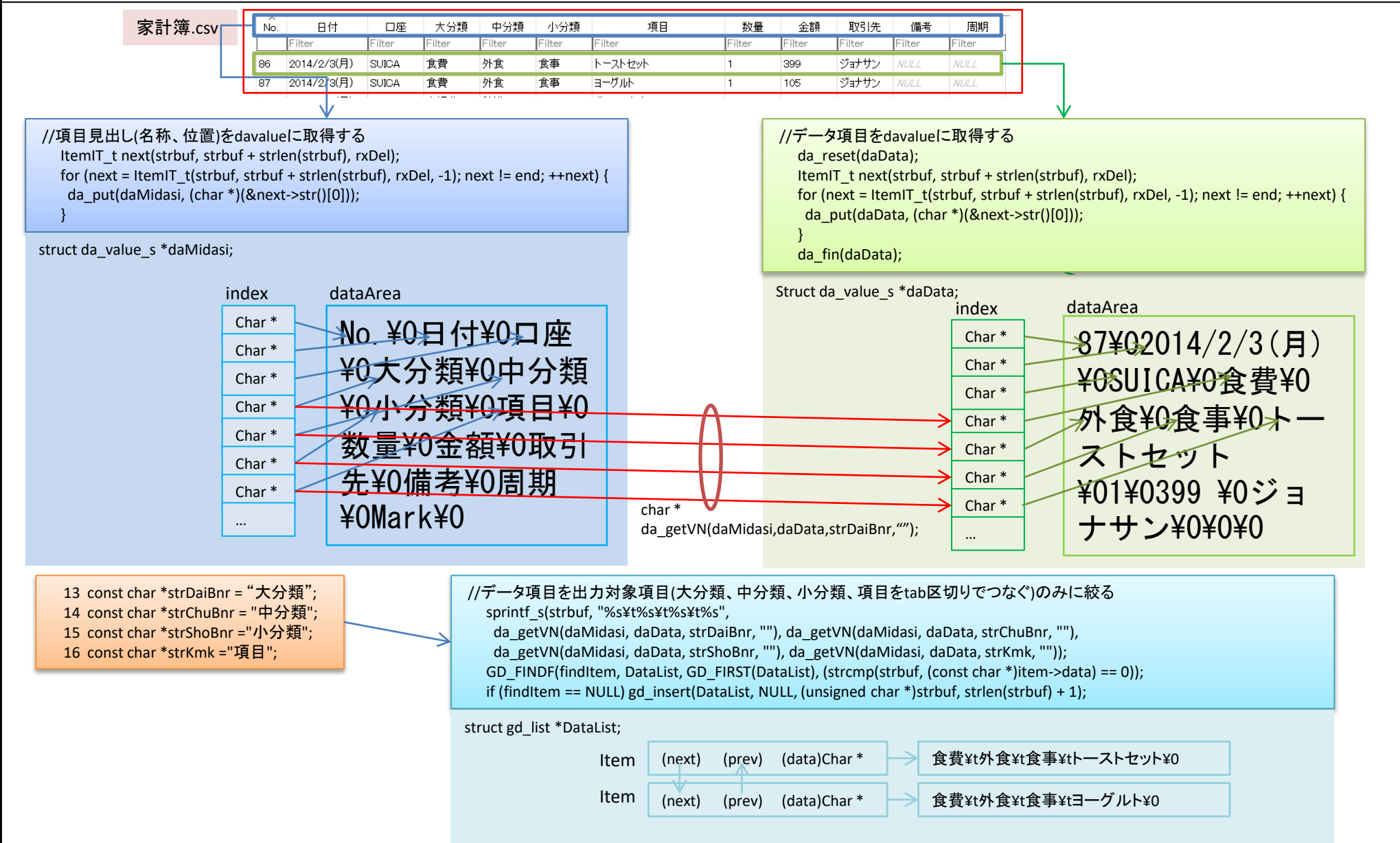
値
{ size=53 }
0x000001be9fdb6010「食費\外食\食事\トーストセット」
0x000001be9fdb6c10「食費\外食\食事\ヨーグルト」

```
139 bool
140 isNewValue(
141     const char *dataStr)
```

モジュール設計仕様書	システム	家計簿.csv	家計簿入力補助・履歴検索	作成日	2020/5/18
補足	モジュール	Visual Studio C++ - CSV - Console	家計簿支出検索(家計簿入力補助) (C++-CSV-コンソール)	作成者	hosiynamakaze
DFD:⑦-1 家計簿支出検索(家計簿入力補助) Visual Studio 2017 C++-CSV コンソールアプリ実装 (map, vector, list版)					
<div><div><div>1 //UTF8 \$Workfile: srchTextCPPConCSV.cpp \$\$Revision: 4 \$\$Date: 20/05/18 19:30 \$</div><div>2 //\$NoKeywords: \$</div><div>3</div><div>4 #include <iostream></div><div>5 #include <fstream></div><div>6 #include <regex></div><div>7 #include <map></div><div>8 #include <vector></div><div>9 #include <list></div><div>10</div><div>11 using namespace std;</div><div>12</div><div>13 const char *strDaiBnr = "大分類"; //出力(使用)する項目の見出し</div><div>14 const char *strChuBnr = "中分類";</div><div>15 const char *strShoBnr = "小分類";</div><div>16 const char *strKmk = "項目";</div><div>17</div><div>18 typedef std::map<char *, int> ItemMap_t; //(見出し,出現順位)を保管する(CSVの全ての見出し)</div><div>19 ItemMap_t ItemMap;</div><div>20 ItemMap_t::iterator ItemMapIT;</div><div>21</div><div>22 vector<char *> rowVec; //データ項目を出現位置順に保管する(行ごとに一時使用)</div><div>23 vector<char *>::iterator rowIT;</div><div>24</div><div>25 list<char *> DataList; //出力および検索に使用する項目をユニークにして保持する</div><div>26 list<char *>::iterator DataListIT;</div><div>27</div><div>28 char *strValue(const char *itemNm); //rowVec,ItemMapから見出しに対応するデータ値を取り出す</div><div>29 bool isNewValue(const char *dataStr); //DataList上でデータ値(出力する大分類～項目)をユニークにするための判定</div><div>30</div><div>31</div><div>32 int main()</div><div>33 {</div><div>34 const char *csvFlNm = "G:/Projects/srchTextCPPCSV/x64/Debug/家計簿.csv";</div><div>35 ifstream* ifCSV = new ifstream;</div><div>36 int csvRecCnt = 0;</div><div>37</div><div>38 typedef regex_token_iterator<const char *> ItemIT_t; //CSVファイルの行ごとにtabで分割する</div><div>39 ItemIT_t::regex_type rxDel("\t");</div><div>40 ItemIT_t end;</div><div>41 cmatch narrowMatch;</div><div>42</div><div>43 char srchText[128]; //検索文字列</div><div>44 char strbuf[1024]; //汎用文字列</div><div>45 int i; //汎用インデックス</div><div>46</div><div>47 /* ----- procedure */</div><div>48</div><div>49 //CSVファイルを読み込む</div><div>50 cout << "家計簿検索 読み込み...\n";</div><div>51 ifCSV->open(csvFlNm);</div><div>52 if (ifCSV->fail()) {</div><div>53 cout << "ERROR: Cannot open file ." << csvFlNm << endl;</div><div>54 goto exit_procedure;</div><div>55 }</div><div>56...以下略</div></div><div><div>srchTextCPPConCSV_mapNvector.cpp</div><div>Revision: 4 までMap,Vector版</div></div></div>					

モジュール設計仕様書	システム	家計簿.csv	家計簿入力補助・履歴検索	作成日	2025/1/25
補足	モジュール	Visual Studio C++ - CSV - Console	家計簿支出検索(家計簿入力補助) (C++-CSV-コンソール)	作成者	hosiynamakaze

DFD:⑦-1' 家計簿支出検索(家計簿入力補助) Visual Studio C++-CSV コンソールアプリ実装(CSVファイルを読み込む概念図) (davalue, gdlist版)



モジュール設計仕様書	システム	家計簿.csv	家計簿入力補助・履歴検索	作成日	2025/1/22
補足	モジュール	Visual Studio C++ - CSV - Console	家計簿支出検索(家計簿入力補助)(C++-CSV-コンソール)	作成者	hosiynamakaze
DFD:⑦-1' 家計簿支出検索(家計簿入力補助) Visual Studio C++-CSV コンソールアプリ実装(davalue, gdlist版)					
<div>1 static char sccsid[] =</div> <div>2 "@(#)Workfile: srchTextCPPConCSV.cpp \$Revision: 8 \$Date: 20/06/06 15:11 \$NoKeywords: \$";</div> <div>3 /*****</div> <div>4 *</div> <div>5 * srchTextCPPConCSV:CSVファイルを取り込み、正規表現で検索する</div> <div>6 * (davalue,gdlistをC++化した)</div> <div>7 *</div> <div>8 *****/</div> <div>9</div> <div>10 #include <iostream></div> <div>11 #include <fstream></div> <div>12 #include <regex></div> <div>13 #include "davalue.h"</div> <div>14 #include "gdlist.h"</div> <div>15</div> <div>16 using namespace std;</div> <div>17</div> <div>18 void regChrEsc(unsigned char *str);</div> <div>19</div> <div>20</div> <div>21 int main()</div> <div>22 {</div> <div>23 const char *strDaiBnr = "大分類";//出力(使用)する項目の見出し</div> <div>24 const char *strChuBnr = "中分類";</div> <div>25 const char *strShoBnr = "小分類";</div> <div>26 const char *strKmk = "項目";</div> <div>27</div> <div>28 const char *csvFlnm = "N:/CMD/家計簿/家計簿.csv";</div> <div>29 ifstream* ifCSV = new ifstream;</div> <div>30 int csvRecCnt = 0;</div> <div>31</div> <div>32 struct da_value_s *daMidasi; //見出しを出現位置順に保管する(csvの全ての見出し)</div> <div>33 struct da_value_s *daData; //データ項目を出現位置順に保管する(行ごとに一時使用)</div> <div>34</div> <div>35 struct gd_list *DataList; //出力および検索に使用するデータ項目をユニークにして保持する</div> <div>36 /* mode:DUP,key:NULL,data:str(大分類、中分類、小分類、項目をtab区切りでつなぐ) */</div> <div>37 struct gd_list *findItem; //データ項目</div> <div>38</div> <div>39 typedef regex_token_iterator<const char*> ItemIT_t; //CSVファイルの行ごとにtabで分割する</div> <div>40 ItemIT_t::regex_type rxDel("\t");</div> <div>41 ItemIT_t end;</div> <div>42</div> <div>43 cmatch narrowMatch;</div> <div>44</div> <div>45 unsigned char srchText[128]; //検索文字列</div> <div>46 char strbuf[1024]; //汎用文字列</div> <div>47</div> <div>48 /* ----- procedure */</div> <div>49</div> <div>50 daMidasi = da_init(1024);</div> <div>51 daData = da_init(1024);</div> <div>52 DataList = gd_init(U_DUP);</div> <div>53</div> <div>54 //CSVファイルを読み込む</div> <div>55 cout << "家計簿検索 読み込み[" << csvFlnm << "]... ";</div> <div>56 ifCSV->open(csvFlnm);</div> <div>57 if (ifCSV->fail()) {</div> <div>58 cout << "ERROR: Cannot open file : " << csvFlnm << endl;</div> <div>59 goto exit_procedure;</div> <div>60 }</div> <div>61...以下略</div>					
<div>srchTextCPPConCSV_davalueNgdlist.cpp</div> <div>Revision: 5 よりdavalue,gdlist版</div>					