

APPENDIKS 23 : Indhold af OnDutyPlanner. DLL
Version 1.0.1

Indhold af OnDutyPlanner. DLL





Revision kontrol:

Version	Dato	Beskrivelse af ændringer	Leveret til
1.0.0	23/03/07	OnDutyPlanner	
1.0.1	01/01/09	Nyt layout	

Versionsnr. ændres efter følgende regler:

Version 1.0.0 er det færdige dokument.

Version 1.0.x er mindre rettelser (af topografisk karakter, og små tilføjelser.)

Version 1.y.0 bliver benyttet væsentlige nye afsnit tilføjes.

Version 2.0.0 bliver benyttet ved næste omskrivning.



Indhold

Indhold	3
SKIFT90-DLL'er.....	4
PMCHAIN: Bibliotek over dobbelt lænket liste	4
Datastruktur	4
Rutiner:	4
PMCHNREC: Bibliotek over dobbelt lænket liste	6
Datastruktur:	6
Rutiner:	6
PMDATTIM: Bibliotek over datobehandling	8
Konstanter:	8
Rutiner:	8
PMLINWRT: Bibliotek over formateret filudskrivning	10
Konstater:	10
Datastruktur:	11
Rutiner:	12
PMSET: Bibliotek over særutiner	14



OnDutyPlanner-DLL'er

Efterfølgende vil nedenfor nævnte DDL'er kort beskrives, idet hovedstrukturer samt eksporterede funktioner beskrives.

PMCHAIN: Bibliotek over dobbelt lænket liste

Datastruktur:

```
typedef struct ZCHAIN {  
    struct ZCHAIN      *pzchPrev;  
    PSZ                pszKey;  
    struct ZCHAIN      *pzchNext;  
} ZCHAIN;
```

```
typedef ZCHAIN *PZCHAIN;
```

Rutiner:

Indsættelse af element i den lænkede liste

```
PZCHAIN ChainInsElm(PZCHAIN*,USHORT);
```

Sletning af element i den lænkede liste

```
PZCHAIN ChainDelElm(PZCHAIN*,PZCHAIN);
```

Slet alt i lænket liste

```
USHORT ChainDelAll(PZCHAIN*);
```

Indeksindsættelse elements buffer



USHORT ChainInsKey(PZCHAIN,USHORT,PSZ,USHORT);

Indeksudtræk fra elements buffer

USHORT ChainGetKey(PZCHAIN,USHORT,PSZ,USHORT);



PMCHNREC: Bibliotek over dobbelt lænket liste

Datastruktur:

```
typedef struct ZCHNREC {  
    struct ZCHNREC *pzchPrev;  
    PVOID          pData;  
    struct ZCHNREC *pzchNext;  
} ZCHNREC;  
  
typedef ZCHNREC      *PZCHNREC;
```

Rutiner:

Indsættelse af element i dobbelt lænket liste

```
PZCHNREC ChnRecInsRec(PZCHNREC*,PVOID,USHORT);
```

Erstat element i dobbelt lænket liste

```
USHORT ChnRecModRec(PZCHNREC,PVOID,USHORT);
```

Modtag element fra dobbelt lænket liste

```
USHORT ChnRecGetRec(PZCHNREC,PVOID,USHORT);
```

Slet element fra dobbelt lænket liste

```
PZCHNREC ChnRecDelRec(PZCHNREC*,PZCHNREC);
```

Slet alle elementer i dobbelt lænket liste

```
USHORT ChnRecDelAll(PZCHNREC*);
```



Modtag antallet af elementer i dobbelt lænket liste

USHORT ChnRecNumRec(PZCHNREC,PLONG);



PMDATTIM: Bibliotek over datobehandling

Konstanter:

INMIN Værdi svarende til et min.

INHOU Værdi svarende til en time.

INDAY Værdi svarende til en dag.

INWEK Værdi svarende til en uge.

Rutiner:

Initialisering af PMDATTIM DLL

```
USHORT InitDatTim(void);
```

Konverter fra SKIFT90 datoformat til SQL format

```
USHORT CFS90DateTime(USHORT,SZ_15,SZ_9,SZ_27);
```

```
USHORT CFS90Date(SZ_15,SZ_27,SZ_27);
```

```
USHORT CFS90Time(SZ_9,SZ_27,SZ_27);
```

Konverter fra SQL til SKIFT90 datoformat

```
USHORT CTS90DateTime(SZ_15,SZ_9,SZ_27);
```

```
USHORT CTS90Date(SZ_15,SZ_27);
```

```
USHORT CTS90Time(SZ_9,SZ_27);
```

Find tidsdiff. mellem 2 tidsangivelser, hhv. SKIFT90 og SQL

```
USHORT DiffS90(SZ_15,SZ_9,SZ_15,SZ_9,double *);
```

```
USHORT DiffSQL(SZ_27,SZ_27,double *);
```




Modificer en SQL tidsangivelse med et givet antal sekunder

USHORT ModifyDateTime(SZ_27, LONG);

Konverter en SQL tidsangivelse fra/til sekunder

USHORT SQLToSec(time_t *, SZ_27);

USHORT SecToSQL(time_t, SZ_27);

Udfra en SQL tidsangivelse bestem antal uger i sidste og indeværende år

USHORT SQLFindNoWeeks(SZ_27, PUSHORT, PUSHORT);

Udfra en SQL tidsangivelse bestem år, ugenummer og ugedag

USHORT SQLToWeekNo(SZ_DT, USHORT*, USHORT*, USHORT*);

Udfra år, ugenummer og ugedag bestem SQL tidsangivelser til hhv. ugens startdato, ugedags dato samt ugens slutdato

USHORT SQLFromWeekNo(USHORT, USHORT, USHORT, SZ_DT[3]);

Udfra basisår (hvis intet angivet i år), antal år til modifikation af basisår og ugenummer bestem SQL tidsangivelser

CFWeekNo(USHORT, USHORT, PSZ, SZ_DT[3]);

Initialisering af DLL

USHORT InitDatTim(void);



PMLINWRT: Bibliotek over formateret filudskrivning

Konstater:

LINWRT_MAX_ENTRIES: Max antal samtidige aktive filer.

PMLINWRT_FORMFEED: Formfeed karakter.

Formatting codes to be translated:

PMLINWRT_USER / U

PMLINWRT_DATE / D

PMLINWRT_TIME / T

PMLINWRT_PAGE / P

Codes formatting a cell:

LINE_CENTER

LINE_RIGHT_JUSTIFY

LINE_LEFT_JUSTIFY

Codes formatting a LINWRTWrite call:

LINE_NORMAL

LINE_SPLIT

LINE_EMPTY

LINE_FORMAT

LINE_UNFORMAT



Datastruktur:

Formatting structure:

```
typedef struct ZLWFMT
{
    USHORT usPagesize;
    USHORT usHeaderSize;
    USHORT usSubHeaderSize;
    USHORT usFooterSize;
    USHORT usStatusLineLength;
    USHORT usCellType;
    CHAR    szFixedStatusLine[MAX_LINE_SIZE];
    CHAR    szFormat[MAX_LINE_SIZE];
    CHAR    szHeaderLine1[MAX_LINE_SIZE];
    CHAR    szHeaderLine2[MAX_LINE_SIZE];
    CHAR    szHeaderLine3[MAX_LINE_SIZE];
    CHAR    szHeaderLine4[MAX_LINE_SIZE];
    CHAR    szHeaderLine5[MAX_LINE_SIZE];
    CHAR    szSubHeaderLine1[MAX_LINE_SIZE];
    CHAR    szSubHeaderLine2[MAX_LINE_SIZE];
    CHAR    szSubHeaderLine3[MAX_LINE_SIZE];
    CHAR    szSubHeaderLine4[MAX_LINE_SIZE];
    CHAR    szSubHeaderLine5[MAX_LINE_SIZE];
    CHAR    szFooterLine1[MAX_LINE_SIZE];
    CHAR    szFooterLine2[MAX_LINE_SIZE];
    CHAR    szFooterLine3[MAX_LINE_SIZE];
    CHAR    szFooterLine4[MAX_LINE_SIZE];
    CHAR    szFooterLine5[MAX_LINE_SIZE];
} ZLWFMT;
```



Rutiner:

Initialiser PMLINWRT DLL

USHORT LinWrtStartUp(void);

Initialisering af indgang, idet ID for denne modtages

USHORT LinWrtInit(PUSHORT,PSZ,PSZ);

Sæt formatet til denne indgang

USHORT LinWrtFormat(USHORT,PZLWFMT);

Udskriv linie til fil

USHORT LinWrtWrite(USHORT,USHORT,PSZ);

Udskriv streng til indekseret buffer

USHORT LinWrtBufIns(PSZ,USHORT,PSZ);

Terminer denne indgang

USHORT LinWrtEnd(USHORT);

Formater linie ud fra indekseret buffer

USHORT LinWrtFormatLine(USHORT, PSZ, PSZ, PSZ);

Udskriv footer på filside

USHORT LinWrtFooter(USHORT);

Udskriv header på filside

USHORT LinWrtHeader(USHORT);

Udskriv sub header på filside

OnDutyPlanner
Teknisk Dokumentation
Appendiks 23 ver.1.0.1
01-01-2009



USHORT LinWrtSubHeader(USHORT);

Initialiser DLL

USHORT LinWrtStartUp(void);

Udskriver en eller flere karakterer

USHORT LinWrtWriteChar(USHORT, ...);

Reset linienummeret

USHORT LinWrtSetLineNo(USHORT,USHORT);

Reset sidenummeret

USHORT LinWrtSetPageNo(USHORT, USHORT);

Modtag linienummeret

USHORT LinWrtGetLineNo(USHORT);

Modtag sidenummeret

USHORT LinWrtPageLineNo(USHORT);

Foretag sideskift

USHORT LinWrtNewPage(USHORT);



PMSET: Bibliotek over sætrutiner

Tøm et sæt

USHORT SetEmpty(USSET);

Bestem et sæt som fællesmængden mellem 2 sæt

USHORT SetUnion(USSET, USSET, USSET);

Bestem et sæt som foreningsmængden mellem 2 sæt

SHORT SetInter(USSET, USSET, USSET);

Adder et element til et sæt

USHORT SetAdd(USSET,USELMNT);

Slet et element fra et sæt

USHORT SetDel(USSET, USELMNT);

Bestem om et element er medlem af et sæt

USHORT SetMember(USSET,USELMNT);

Bestem næste fri elementposition i et sæt

USHORT SetNextUnused(USSET,USELMNT *);

Udskriv et sæt

USHORT SetPrint(USSET,CHAR *);