

## NSSDCA Document B56521-000A

### Apollo ARCSAV Level 0 (Raw Binary) File Processing Programs and Supplementary Input Files

Program <sup>1</sup>	Description	Main Input Files	Supplementary Input Files	Output Files <sup>2,3</sup>	Last Revision Date
renamefj.f	Rename JBI files for NSSDCA	e*.tap	filenamecrj.txt	arcsav_ap*_*P*_j	12/10/15
renamefd.f	Rename DBDS files for NSSDCA	28c1.*.bcd	filenamecrd.txt	arcsav_ap*_*P*_d*	6/11/16
renamefk.f	Rename KDM files for NSSDCA	*.tap	filenamecrk.txt	arcsav_ap*_*P*_k	11/19/16
renamefk2.f	Rename KDM files for NSSDCA, v. 2	*.tap	filenamecrk2.txt	arcsav_ap*_*P*_k	4/30/17
db-to-jbi.2.c	Format conversion from DBDS to JBI	28c1.*.bcd		28c1.*.jbi	3/25/14
kat_to_jbi.c	Format conversion from KDM to JBI	*.tap		*.tap.jbi	11/27/16
renamefdj.f	Rename JBI format DBDS files to NSSDCA	28c1.*.jbi	filenamecrd.txt	arcsav_ap*_*P*_dj*	6/11/16
renamefkj.f	Rename JBI format KDM files to NSSDCA	*.tap.jbi	filenamecrk.txt	arcsav_ap*_*P*_kj	11/19/16
renamefkj2.f	Rename JBI format KDM files to NSSDCA, v. 2	*.tap.jbi	filenamecrk2.txt	arcsav_ap*_*P*_kj	4/30/17

<sup>1</sup> These Fortran (\*.f) and C language (\*.c) programs are provided by the data provider, Yosio Nakamura, for documentation purposes only. The programs are provided in alphabetical order on pages 2-13 of this document. The Supplementary Input Files used by these programs are listed in alphabetical order on pages 14-24.

<sup>2</sup> The Output Files named “arcsav\_ap\*” are the renamed vendor-extracted ARCSAV tape files that are grouped into five Level 0 (raw binary) data collections by Apollo mission and archived at the NASA Space Science Data Coordinated Archive (NSSDCA) under these collection identifiers: PSPG-00912 (Apollo 12); PSPG-00913 (Apollo 14); PSPG-00914 (Apollo 15); PSPG-00915 (Apollo 16); and PSPG-0916 (Apollo 17). The archived file names are cross-referenced to the Input original vendor-based file names in NSSDCA document B56516, <https://nssdc.gsfc.nasa.gov/misc/documents/b56516.pdf>. Level 0 data format is described in NSSDCA document B56515, <https://nssdc.gsfc.nasa.gov/misc/documents/b56515.pdf>.

<sup>3</sup> The Output files named “28c1.\*.jbi” and “\*.tap.jbi” are the files that vendors DBDS and KDM extracted from the ARCSAV tapes. The C-language programs db-to-jbi.2.c and kat\_to\_jbi.c convert the vendor-specific file formats to the format used by vendor JBI’s format.

## db-to-jbi.2.c

```
/*
 * db-to-jbi.c 10/2013 mark wiederspahn
 *
 * v2 3/2014 mw    add record number to unexpected size print
 *                at last record, also report unexpected size
 *                check parity, report such each block
 *
 * change format of 7 track decoded files:
 * change databank files to jbi files
 * if sign bit is set, this is the first byte in a record.
 * remember the byte count, and emit that, then the record,
 * then the byte count again, similar to how jbi did it.
 *
 * the byte counts are little endian i*4
 *
 * db-to-jbi < databank-file > jbi-file
 */

#include <stdio.h>

main( int argc, char *argv[] )
{
    char  buf[100000];
    char  *bp;
    int   in;
    int   count;      /* block length */
    int   nrec; /* count of blocks */
    int   nerr; /* number of bad block lengths */
    int   nperr;      /* number of parity errors per block */
    int   nptot;      /* total parity errors */
    int   nbytes;     /* total bytes read == sum of counts */

    bp = buf;

    nrec = 0;
    nerr = 0;
    nperr = 0;
    nptot = 0;
    nbytes = 0;
    while( (in=getchar()) != EOF )
    {
        if( (in & 0x80) != 0 ) /* start of a record */
        {
            if( bp != buf ) /* if any data already seen? */
            {
                count = (int)(bp - buf );
                if( count != 19200 )
                {
                    nerr++;
                    fprintf( stderr, "unexpected len %d at rec
%d\n", count, nrec ); /*
                }
                nbytes += count;
                fprintf( stderr, "%d %d %d\n", nbytes, count, nperr );
            }
        }
    }
}
```

```

/*          write( fileno(stdout), &count, 4 );
           write( fileno(stdout), buf, count );
           write( fileno(stdout), &count, 4 );
*/
        }
        nrec++;
        nptot += nperr;
        nperr = 0;
        bp = buf;
    }
    nperr += 1 - is_good_parity( in & 0x7f );
    *bp++ = in & 0x7f; /* zap sign bit if there */
}
if( bp != buf )
{
    count = (int)(bp - buf );
    if( count != 19200 )
    {
        nerr++;
        fprintf( stderr, "unexpected len %d at rec %d\n", count,
nrec );
    }
    nbytes += count;
    fprintf( stderr, "%d %d %d\n", nbytes, count, nperr );
/*
write( fileno(stdout), &count, 4 );
write( fileno(stdout), buf, count );
write( fileno(stdout), &count, 4 );
*/
}
nptot += nperr;
fprintf( stderr, "%d total len err %d perrs\n", nerr, nptot );
}

/* is_good_parity() returns 1 if 6 data + parity is correct */

int is_good_parity( in )
{
    int i, n;

    n = 0;
    for( i=0; i<7; i++ )
    {
        if( in & 1 ) n++;
        in = in >> 1;
    }
    return n%2 == 1;
}

```

## kat\_to\_jbi.c

```
/*
 * kat_to_jbi.c read a Katalyst 7 track tape image and make a JBI tape image,
 *
 * updated 2016/11/27 mw - clarify some descriptions. No code changes.
 *
 * The Katalyst input does not have any parity bits, only low 6 bits.
 * Add parity bits as we go. The two 48byte headers are even parity,
 * all the other data records are odd parity.
 *
 * We assume a little-endian computer; this only affects the 3 header
 * integers.
 *
 * input: repeat forever: (record header,data block)
 * record header: 12 byte header: is_eof, leof, leod all 32 bit little-endian
 * ints:
 *   is_eof is 0 if data follow, 1 if this is an eof. Not sure of leof/leod
 *   values if so.
 *   leof is byte offset of the start of the descriptor for previous record
 *   (0 if beginning)
 *   leod is byte offset of the start of the descriptor for the next record
 *   (0 if end)
 *   we could read this a struct, but it is simple to treat it as 3 4-byte
 *   chunks.
 * data block: tape record, 6 bits in low bits of each byte, top 2 bits 0,
 *   so no parity.
 *
 * output JBI format: byte count before and after each data block (32 bit
 * little endian ints).
 * eof marks are zero byte records. JBI data block includes bit 6 parity,
 * and flags bad hardware detected parity with bit 8 set
 */

#include <stdio.h>
#include <stdlib.h>

main()
{
    int    is_eof, leof, leod;
    int    nb, nt;
    int    nr, nw;
    int    zero=0;
    char   buf[20000]; /*expect only 19200 max */

    nt = 0;           /* total bytes read so far */
    nr = 0;           /* records read so far */

    while( read( 0,&is_eof, 4 ) == 4 )
    {
        if( read( 0, &leof, 4 ) != 4 ) exit( 1 );
        if( read( 0, &leod, 4 ) != 4 ) exit( 1 );

/*
        fprintf( stderr, "%d %d %d %d\n", is_eof, leof, leod, nt ); */

        nr ++;
    }
}
```

```

nt += 12;
nb = leod - nt;

if( is_eof )
{
    fprintf( stderr, "EOF at %d\n", nr );
    write( 1, &zero, 4 );
}
else
{
    if( nb > sizeof( buf ) )
    {
        fprintf( stderr, "NB too large increase buf to %d\n",
nb );
        exit( 1 );
    }

    if( read( 0, buf, nb ) != nb )
    {
        fprintf( stderr, "read error %d %d\n", nr, nb );
        perror( "read" );
        exit( 1 );
    }

    nt += nb;

    if( nr > 2 && nb != 19200 ) fprintf( stderr, "NB abnormal
%d %d\n", nr, nb );

    if( nr > 2 )
    {
        setodd( buf, nb );
    }
    else
    {
        seteven( buf, nb );
    }

    nw=write( 1, &nb, 4 );
    nw=write( 1, buf, nb );
    nw=write( 1, &nb, 4 );
}
}
}

```

```

setodd( char *buf, int nb )
{
    int i, j, n;
    char c;

    for( i=0; i<nb; i++ )
    {
        n = 0;
        for( j=0; j<6; j++ )
        {

```

```

/*          fprintf( stderr,"%d %d %x\n", i, j, (int)(buf[i] >> j) );
*/
    if( (buf[i] >> j) & 1 )
    {
        n++;
    }

    }
    if( ! (n&1) )          /* if it isn't odd */
    {
        buf[i] |= 0x40;    /* make it odd */
    }
/*          fprintf( stderr, "n %d %x %x\n", n, (int)c, (int)buf[i] ); */
    }
}
seteven( char *buf, int nb )
{
    int  i,j,n;

    for( i=0; i<nb; i++ )
    {
        n = 0;
        for( j=0; j<6; j++ )
/*          fprintf( stderr,"%d %d %x\n", i, j, (int)(buf[i] >> j) );
*/
        if( (buf[i] >> j) & 1 )
        {
            n++;
        }

    }
    if( (n&1) )          /* if it isn't even */
    {
        buf[i] |= 0x40;    /* make it even */
    }
/*          fprintf( stderr, "n %d %x\n", n, (int)buf[i] ); */
    }
}

```

## reanamefd.f

```
      program reanamefd
c copy and rename DBDS files for NSSDC
c **11-Dec-15**rev. 10-Jun-16**
      character tape*6,filed*22,filen*29
      data filen/'arcsav_ap**_1975***_*****_d*'/
c
      open(1,file='../lists/filenamecrd.txt',status='old')
      ld=0
10     read(1,*,end=90) ns,nd,tape,filed
      write(*,*) ns,nd,' ',tape,' ',filed
      write(filn(10:11),'(i2)') ns
      write(filn(17:19),'(i3)') nd
      if (filn(17:17).eq.' ') filn(17:17)='0'
      write(filn(21:26),'(a6)') tape
      if (filed(7:7).eq.'/') then
         is=8
      else
         is=9
      end if
      do i=15,20
         if (filed(i:i).eq.'b') go to 11
      end do
11     ie=i+2
      write(*,*) is,ie
      if (nd.ne.ld) then
         filn(29:29)='1'
      else
         filn(29:29)='2'
      end if
      write(*,*) filed(is:ie)//' '//filn
      call system('cp -p ../from_db//filed(1:ie)//' ../files2nssdc')
      call system
&      ('mv ../files2nssdc//filed(is:ie)//' ../files2nssdc//filen)
      ld=nd
      go to 10
90    close(1)
      stop
      end
```

## reanamefdj.f

```
      program reanamefdj
c copy and rename JBI-format DBDS files for NSSDC
c **10-Jun-16**
      character tape*6,filed*22,filedj*14,cp2*16,filen*30
      data cp2/'cp -p ../in_jbi//',filen/'arcsav_ap**_1975***_*****_dj*'/
c
      open(1,file='../lists/filenamecrd.txt',status='old')
      ld=0
10     read(1,*,end=90) ns,nd,tape,filed
      do i=7,8
         if (filed(i:i).eq.'/') go to 11
      end do
11     js=i+1
      do i=js+7,20
         if (filed(i:i).eq.'b') go to 12
      end do
12     lf=i-js+3
         je=i-1
         filedj(11:14)= '      '
         filedj(1:lf)=filed(js:je)//'jbi'
         write(*,*) ns,nd,' ',tape,' ',filedj
         write(filn(10:11),'(i2)') ns
         write(filn(17:19),'(i3)') nd
         if (filn(17:17).eq.' ') filn(17:17)='0'
         write(filn(21:26),'(a6)') tape
         if (filed(7:7).eq.'/') then
            ns=8
         else
            ns=9
         end if
         if (nd.ne.ld) then
            filn(30:30)='1'
         else
            filn(30:30)='2'
         end if
         write(*,*) filedj//' '//filen
         call system(cp2//filedj//' ../files2nssdc')
         call system
&      ('mv ../files2nssdc//filedj//' ../files2nssdc//filen)
         ld=nd
         go to 10
90     close(1)
         stop
         end
```



## reannamefj.f

```
      program reannamefj
c copy and rename JBI files for NSSDC
c **9-Dec-15
      character tape*6,filej*20,tojf*21,jcap*16,filen*28
      data tojf/'../jbi2/NASAPATRICK2/'/
      data filen/'arcsav_ap**_1975***_*****_j'/
c
      open(1,file='../lists/filenamecrj.txt',status='old')
10    read(1,*,end=90) ns,nd,tape,filej
      jcap='E'//filej(2:16)
      do i=13,16
         ic=ichar(jcap(i:i))
         if (ic.ge.97) jcap(i:i)=char(ic-32)
      end do
      write(filen(10:11),'(i2)') ns
      write(filen(17:19),'(i3)') nd
      if (filen(17:17).eq.' ') filen(17:17)='0'
      write(filen(21:26),'(a6)') tape
      write(*,*) jcap//'/'/filej//'/ '/filen
      call system('cp -p '//tojf//jcap//'/filej//' ../files2nssdc')
      call system('mv ../files2nssdc//filej//' ../files2nssdc//filen)
      go to 10
90   close(1)
      stop
      end
```

## reanamefk.f

```
      program reanamefk
c copy and reaname KDM files for NSSDC
c **18-Nov-16**
      character tape*6,filek*29,filen*28
      data filen/'arcsav_ap**_1975***_*****_k'/
c
      open(1,file='../lists/filenamecrk.txt',status='old')
      ld=0
10    read(1,*,end=90) ns,nd,tape,nx,filek
      write(*,*) ns,nd,' ',tape,filek
      write(filn(10:11),'(i2)') ns
      write(filn(17:19),'(i3)') nd
      if (filn(17:17).eq.' ') filn(17:17)='0'
      write(filn(21:26),'(a6)') tape
      call system ('cp -p ../as_received/TAP_10'
& //filek(1:nx)//tape(3:6)//'.tap ../files2nssdc')
      if (nx.eq.22) then
          n1=10
      else
          n1=8
      end if
      call system ('mv ../files2nssdc/'
& //filek(n1:nx)//tape(3:6)//'.tap ../files2nssdc/'//filen)
      ld=nd
      go to 10
90    close(1)
      stop
      end
```

## renamefk2.f

```
      program renamefk2
c copy and rename KDM files for NSSDC v.2
c **13--Apr-17**
      character tape*6,filek*49,filen*28
      data filek/'H012873*****_ALL.tap'/
&      ,filen/'arcsav_ap**_1975**_*****_k'/
c
      open(1,file='../lists/filenamecrk2.txt',status='old')
      ld=0
10     read(1,*,end=90) ns,nd,tape,nx,filek(8:40)
      write(*,*) ns,nd,' ',tape,' ',filek
      write(filn(10:11),'(i2)') ns
      write(filn(17:19),'(i3)') nd
      if (filn(17:17).eq.' ') filn(17:17)='0'
      write(filn(21:26),'(a6)') tape
&     call system ('cp -p ../as_received/TAP_04_26_17/'//filek
//' ../files2nssdc/run2')
&     call system ('mv ../files2nssdc/run2/'//filek//
' ../files2nssdc/run2/'//filen)
      ld=nd
      go to 10
90    close(1)
      stop
      end
```

## reanamefkj.f

```
      program reanamefkj
c copy and reaname JBI-format KDM files for NSSDC
c **19-Nov-16**
      character tape*6,filen*29
      data filen/'arcsav_ap**_1975***_*****_kj'/
c
      open(1,file='../lists/filenamecrk.txt',status='old')
      ld=0
10    read(1,*,end=90) ns,nd,tape
      write(*,*) ns,nd,' ',tape
      write(filen(10:11),'(i2)') ns
      write(filen(17:19),'(i3)') nd
      if (filen(17:17).eq.' ') filen(17:17)='0'
      write(filen(21:26),'(a6)') tape
      call system ('cp -p ../links/K'//tape(3:6)//' ../files2nssdc')
      call system ('mv ../files2nssdc/K'//tape(3:6)
&      //' ../files2nssdc/'//filen)
      ld=nd
      go to 10
90    close(1)
      stop
      end
```

## reanamefkj2.f

```
      program reanamefkj2
c copy and reaname JBI-format KDM files for NSSDCv.2
c **30-Apr-17**
      character tape*6,filen*29
      data filen/'arcsav_ap**_1975***_*****_kj'/
c
      open(1,file='../lists/filenamecrk2.txt',status='old')
      ld=0
10     read(1,*,end=90) ns,nd,tape
      write(*,*) ns,nd,' ',tape
      write(filen(10:11),'(i2)') ns
      write(filen(17:19),'(i3)') nd
      if (filen(17:17).eq.' ') filen(17:17)='0'
      write(filen(21:26),'(a6)') tape
      call system ('cp -p ../links/K'//tape(3:6)//' ../files2nssdc/run2')
      call system ('mv ../files2nssdc/run2/K'//tape(3:6)
&      //' ../files2nssdc/run2/'//filen)
      ld=nd
      go to 10
90    close(1)
      stop
      end
```

**filenamecrd.txt**

12	93	P04369	"batchs/28c1.13.bcd"
12	94	P04370	"batch2/28c1.14r.bcd"
12	94	P04370	"batch27/28c1.14r2.bcd"
12	95	P04371	"batch2/28c1.15r.bcd"
12	95	P04371	"batch27/28c1.15r1.bcd"
12	96	P04372	"batch2/28c1.16r.bcd"
12	97	P04373	"batch2/28c1.17r.bcd"
12	97	P04373	"batch27/28c1.17r1.bcd"
12	98	P04374	"batch2/28c1.18r.bcd"
12	98	P04374	"batch27/28c1.18r1.bcd"
12	99	P04375	"batch2/28c1.19r.bcd"
12	99	P04375	"batch27/28c1.19r1.bcd"
12	100	P04376	"batch2/28c1.20r.bcd"
12	100	P04376	"batch27/28c1.20r1.bcd"
12	101	P04377	"batch3/28c1.21r.bcd"
12	102	P04378	"batch3/28c1.22r.bcd"
12	103	P04379	"batch3/28c1.23.bcd"
12	104	P04380	"batch3/28c1.24r.bcd"
12	105	P04381	"batch3/28c1.25r.bcd"
12	106	P04382	"batch3/28c1.26.bcd"
12	107	P04383	"batch3/28c1.27.bcd"
12	108	P04384	"batch3/28c1.28.bcd"
12	109	P04385	"batch3/28c1.29.bcd"
12	110	P04386	"batch3/28c1.30.bcd"
12	111	P04387	"batch3/28c1.31.bcd"
12	112	P04388	"batch3/28c1.32.bcd"
12	113	P04389	"batch4/28c1.33.bcd"
12	113	P04389	"batch28/28c1.33r1.bcd"
12	114	P04390	"batch4/28c1.34.bcd"
12	114	P04390	"batch28/28c1.34r1.bcd"
12	115	P04391	"batch4/28c1.35.bcd"
12	116	P04392	"batch4/28c1.36.bcd"
12	116	P04392	"batch28/28c1.36r1.bcd"
12	117	P04393	"batch4/28c1.37.bcd"
12	118	P04394	"batch4/28c1.38.bcd"
12	119	P04395	"batch4/28c1.39.bcd"
12	119	P04395	"batch28/28c1.39r1.bcd"
12	120	P04396	"batch4/28c1.40.bcd"
12	121	P04397	"batch4/28c1.41.bcd"
12	122	P04398	"batch4/28c1.42.bcd"
12	123	P00018	"batch4/28c1.43.bcd"
12	124	P00019	"batch4/28c1.44.bcd"
12	124	P00019	"batch29/28c1.44r1.bcd"
12	125	P00020	"batch4/28c1.45.bcd"
12	126	P00021	"batch4/28c1.46.bcd"
12	127	P00022	"batch4/28c1.47.bcd"
12	128	P00023	"batch4/28c1.48.bcd"
12	129	P00026	"batch4/28c1.49.bcd"
12	130	P00027	"batch4/28c1.50.bcd"
12	130	P00027	"batch29/28c1.50r1.bcd"
12	131	P00028	"batch4/28c1.51.bcd"
12	131	P00028	"batch29/28c1.51r1.bcd"
12	132	P00029	"batch4/28c1.52.bcd"
12	132	P00029	"batch29/28c1.52r1.bcd"
12	133	P00030	"batch6/28c1.56.bcd"
12	133	P00030	"batch27/28c1.56r1.bcd"

12	134	P00031	"batch6/28c1.57.bcd"
12	134	P00031	"batch27/28c1.57r1.bcd"
12	135	P00032	"batch6/28c1.58.bcd"
12	135	P00032	"batch27/28c1.58r1.bcd"
12	136	P00055	"batch6/28c1.59.bcd"
12	136	P00055	"batch27/28c1.59r1.bcd"
12	137	P00075	"batch6/28c1.60.bcd"
12	137	P00075	"batch27/28c1.60r1.bcd"
12	138	P00076	"batch6/28c1.61.bcd"
12	138	P00076	"batch27/28c1.61r1.bcd"
12	139	P00077	"batch6/28c1.62.bcd"
12	139	P00077	"batch27/28c1.62r1.bcd"
12	140	P00081	"batch6/28c1.63.bcd"
12	140	P00081	"batch27/28c1.63r1.bcd"
12	141	P00082	"batch7/28c1.64.bcd"
12	141	P00082	"batch27/28c1.64r1.bcd"
12	142	P00084	"batch7/28c1.65.bcd"
12	142	P00084	"batch27/28c1.65r1.bcd"
12	143	P00104	"batch5/28c1.66.bcd"
12	143	P00104	"batch25/28c1.66r.bcd"
12	144	P00114	"batch5/28c1.67.bcd"
12	144	P00114	"batch25/28c1.67r.bcd"
12	145	P00133	"batch5/28c1.68.bcd"
12	145	P00133	"batch25/28c1.68r.bcd"
12	146	P00135	"batch5/28c1.69.bcd"
12	146	P00135	"batch25/28c1.69r.bcd"
12	147	P00145	"batch5/28c1.70.bcd"
12	147	P00145	"batch25/28c1.70r.bcd"
12	148	P00146	"batch5/28c1.71.bcd"
12	148	P00146	"batch25/28c1.71r.bcd"
12	149	P00150	"batch5/28c1.72.bcd"
12	149	P00150	"batch25/28c1.72r.bcd"
12	150	P00153	"batch5/28c1.73.bcd"
12	150	P00153	"batch25/28c1.73r.bcd"
12	151	P00158	"batch5/28c1.74.bcd"
12	151	P00158	"batch25/28c1.74r.bcd"
12	152	P00164	"batch5/28c1.75.bcd"
12	152	P00164	"batch25/28c1.75r.bcd"
12	153	P02763	"batch8/28c1.96.bcd"
12	154	P02764	"batch8/28c1.97.bcd"
12	155	P02765	"batch8/28c1.98.bcd"
12	156	P02766	"batch8/28c1.99.bcd"
12	157	P02767	"batch8/28c1.100.bcd"
12	157	P02767	"batch29/28c1.100r1.bcd"
12	158	P02768	"batch8/28c1.101.bcd"
12	159	P03783	"batch8/28c1.102.bcd"
12	160	P04325	"batch8/28c1.103.bcd"
12	161	P04326	"batch8/28c1.104.bcd"
12	162	P04327	"batch8/28c1.105.bcd"
12	163	P04328	"batch8/28c1.106.bcd"
12	164	P04329	"batch8/28c1.107.bcd"
12	165	P04330	"batch8/28c1.108.bcd"
12	166	P04331	"batch8/28c1.109.bcd"
12	167	P04332	"batch8/28c1.110.bcd"
12	168	P04333	"batch8/28c1.111.bcd"
12	169	P04334	"batch8/28c1.112.bcd"
12	170	P04335	"batch8/28c1.113.bcd"
12	171	P04336	"batch8/28c1.114.bcd"

12	172	P04337	"batch8/28c1.115.bcd"
12	173	P04338	"batch8/28c1.86.bcd"
12	174	P04339	"batch8/28c1.87.bcd"
12	175	P04340	"batch8/28c1.88.bcd"
12	176	P04341	"batch8/28c1.89.bcd"
12	177	P04342	"batch8/28c1.90.bcd"
12	178	P04343	"batch8/28c1.91.bcd"
12	179	P04344	"batch8/28c1.92.bcd"
12	180	P04345	"batch8/28c1.93.bcd"
12	181	P04346	"batch8/28c1.94.bcd"
12	182	P04347	"batch8/28c1.95.bcd"
14	103	P04409	"batchs/28c1.3.bcd"
14	104	P04410	"batch1/28c1.4.bcd"
14	105	P04411	"batch1/28c1.5.bcd"
14	106	P04412	"batch1/28c1.6.bcd"
14	107	P04413	"batch1/28c1.7.bcd"
14	108	P04414	"batch1/28c1.8.bcd"
14	109	P04415	"batch2/28c1.9.bcd"
14	110	P04416	"batch2/28c1.10r.bcd"
14	111	P04417	"batch2/28c1.11r.bcd"
14	112	P04418	"batch2/28c1.12r.bcd"
14	113	P04419	"batch6/28c1.53.bcd"
14	114	P04420	"batch6/28c1.54.bcd"
14	115	P04421	"batch6/28c1.55.bcd"
15	94	P04431	"trial2/28c1.1.bcd"
15	96	P04433	"trial2/28c1.2.bcd"
15	104	P04441	"batch30/28c1.314.bcd"
15	109	P04446	"batch30/28c1.315.bcd"
15	110	P04447	"batch30/28c1.316.bcd"
15	114	P04451	"batch30/28c1.313.bcd"
15	121	P04458	"batch24/28c1.309.bcd"
15	122	P00940	"batch18/28c1.233.bcd"
15	123	P00941	"batch18/28c1.234.bcd"
15	124	P00947	"batch18/28c1.235.bcd"
15	125	P00949	"batch18/28c1.236.bcd"
15	126	P00976	"batch18/28c1.237.bcd"
15	127	P00977	"batch18/28c1.238.bcd"
15	128	P00979	"batch18/28c1.239.bcd"
15	129	P00980	"batch18/28c1.242.bcd"
15	130	P00981	"batch18/28c1.240.bcd"
15	130	P00981	"batch29/28c1.240r1.bcd"
15	131	P00982	"batch18/28c1.241.bcd"
15	132	P00983	"batch11/28c1.156.bcd"
15	132	P00983	"batch29/28c1.156r1.bcd"
15	133	P00984	"batch11/28c1.157.bcd"
15	133	P00984	"batch29/28c1.157r1.bcd"
15	134	P01084	"batch11/28c1.158.bcd"
15	135	P01117	"batch11/28c1.159.bcd"
15	136	P01232	"batch11/28c1.160.bcd"
15	137	P01270	"batch11/28c1.161.bcd"
15	138	P01271	"batch11/28c1.162.bcd"
15	139	P01272	"batch11/28c1.163.bcd"
15	140	P01314	"batch11/28c1.164.bcd"
15	141	P01315	"batch11/28c1.165.bcd"
15	142	P01316	"batch11/28c1.166.bcd"
15	143	P01317	"batch11/28c1.167.bcd"
15	144	P01318	"batch11/28c1.168.bcd"
15	145	P01319	"batch11/28c1.169.bcd"



15	146	P01320	"batch11/28c1.170.bcd"
15	147	P01321	"batch11/28c1.171.bcd"
15	148	P01322	"batch12/28c1.172.bcd"
15	149	P01323	"batch13/28c1.173.bcd"
15	150	P01324	"batch13/28c1.174.bcd"
15	151	P01325	"batch13/28c1.175.bcd"
15	152	P04520	"batch13/28c1.176.bcd"
15	153	P04521	"batch13/28c1.177.bcd"
15	154	P04522	"batch13/28c1.178.bcd"
15	155	P04523	"batch13/28c1.179.bcd"
15	156	P04524	"batch13/28c1.180.bcd"
15	157	P04525	"batch13/28c1.181.bcd"
15	158	P04526	"batch13/28c1.182.bcd"
15	159	P04527	"batch13/28c1.183.bcd"
15	160	P04528	"batch13/28c1.184.bcd"
15	161	P04529	"batch13/28c1.185.bcd"
15	162	P04530	"batch19/28c1.263.bcd"
15	163	P04531	"batch19/28c1.264.bcd"
15	164	P04532	"batch19/28c1.265.bcd"
15	165	P04533	"batch19/28c1.266.bcd"
15	165	P04533	"batch29/28c1.266r1.bcd"
15	166	P04534	"batch19/28c1.267.bcd"
15	167	P04535	"batch19/28c1.268.bcd"
15	168	P04536	"batch19/28c1.269.bcd"
15	169	P04537	"batch20/28c1.270.bcd"
15	170	P04538	"batch20/28c1.271.bcd"
15	171	P04539	"batch20/28c1.272.bcd"
15	172	P04540	"batch21/28c1.283.bcd"
15	173	P04541	"batch21/28c1.284.bcd"
15	174	P04542	"batch21/28c1.285.bcd"
15	175	P04543	"batch21/28c1.286.bcd"
15	176	P04544	"batch21/28c1.287.bcd"
15	177	P04545	"batch21/28c1.288.bcd"
15	178	P04546	"batch21/28c1.289.bcd"
15	179	P04547	"batch21/28c1.290.bcd"
15	180	P04548	"batch21/28c1.291.bcd"
15	181	P04549	"batch21/28c1.292.bcd"
16	92	P04459	"batch10/28c1.125.bcd"
16	93	P04460	"batch10/28c1.124.bcd"
16	94	P04461	"batch10/28c1.123.bcd"
16	95	P04462	"batch10/28c1.122.bcd"
16	96	P04463	"batch10/28c1.121.bcd"
16	97	P04464	"batch10/28c1.120.bcd"
16	98	P04465	"batch10/28c1.119.bcd"
16	99	P04466	"batch10/28c1.118.bcd"
16	100	P04467	"batch10/28c1.117.bcd"
16	101	P04468	"batch10/28c1.116.bcd"
16	102	P04469	"batch10/28c1.145.bcd"
16	103	P04470	"batch10/28c1.144.bcd"
16	104	P04471	"batch10/28c1.143.bcd"
16	105	P04472	"batch10/28c1.142.bcd"
16	106	P04473	"batch10/28c1.141.bcd"
16	107	P04474	"batch10/28c1.140.bcd"
16	108	P04475	"batch10/28c1.139.bcd"
16	109	P04476	"batch10/28c1.138.bcd"
16	110	P04477	"batch10/28c1.137.bcd"
16	111	P04478	"batch10/28c1.136.bcd"
16	112	P04479	"batch10/28c1.155.bcd"

16	113	P04480	"batch10/28c1.154.bcd"
16	114	P04481	"batch10/28c1.153.bcd"
16	115	P04482	"batch10/28c1.152.bcd"
16	116	P04483	"batch10/28c1.151.bcd"
16	117	P04484	"batch10/28c1.150.bcd"
16	118	P04485	"batch10/28c1.149.bcd"
16	119	P04486	"batch10/28c1.148.bcd"
16	120	P04487	"batch10/28c1.147.bcd"
16	121	P04488	"batch10/28c1.146.bcd"
16	122	P01326	"batch18/28c1.253.bcd"
16	123	P01327	"batch18/28c1.254.bcd"
16	124	P01328	"batch18/28c1.255.bcd"
16	125	P01329	"batch18/28c1.256.bcd"
16	126	P01330	"batch18/28c1.257.bcd"
16	127	P01331	"batch18/28c1.258.bcd"
16	128	P01332	"batch18/28c1.259.bcd"
16	129	P01333	"batch19/28c1.260.bcd"
16	130	P01334	"batch19/28c1.261.bcd"
16	131	P01335	"batch19/28c1.262.bcd"
16	132	P01336	"batch20/28c1.273.bcd"
16	133	P01337	"batch20/28c1.274.bcd"
16	134	P01338	"batch20/28c1.275.bcd"
16	135	P01342	"batch20/28c1.276.bcd"
16	136	P04610	"batch20/28c1.277.bcd"
16	137	P01343	"batch20/28c1.278.bcd"
16	138	P01344	"batch20/28c1.279.bcd"
16	139	P01375	"batch20/28c1.280.bcd"
16	140	P01382	"batch21/28c1.281.bcd"
16	141	P01467	"batch21/28c1.282.bcd"
16	142	P01924	"batch13/28c1.186.bcd"
16	143	P01965	"batch13/28c1.187.bcd"
16	144	P02380	"batch13/28c1.188.bcd"
16	145	P02496	"batch13/28c1.189.bcd"
16	146	P02497	"batch13/28c1.190.bcd"
16	147	P02498	"batch13/28c1.191.bcd"
16	148	P02499	"batch13/28c1.192.bcd"
16	149	P02500	"batch13/28c1.193.bcd"
16	150	P02735	"batch13/28c1.194.bcd"
16	151	P02736	"batch13/28c1.195.bcd"
16	152	P04550	"batch14/28c1.203.bcd"
16	153	P04551	"batch14/28c1.204.bcd"
16	154	P04552	"batch14/28c1.205.bcd"
16	155	P04553	"batch14/28c1.206.bcd"
16	156	P04554	"batch14/28c1.207.bcd"
16	157	P04555	"batch14/28c1.208.bcd"
16	158	P04556	"batch14/28c1.209.bcd"
16	159	P04557	"batch14/28c1.210.bcd"
16	160	P04558	"batch15/28c1.211.bcd"
16	161	P04559	"batch15/28c1.212.bcd"
16	162	P04560	"batch10/28c1.126.bcd"
16	163	P04561	"batch10/28c1.127.bcd"
16	164	P04562	"batch10/28c1.128.bcd"
16	165	P04563	"batch10/28c1.129.bcd"
16	166	P04564	"batch10/28c1.130.bcd"
16	167	P04565	"batch10/28c1.131.bcd"
16	168	P04566	"batch10/28c1.132.bcd"
16	169	P04567	"batch10/28c1.133.bcd"
16	170	P04568	"batch10/28c1.134.bcd"

16	171	P04569	"batch10/28c1.135.bcd"
16	172	P04570	"batch16/28c1.223.bcd"
16	173	P04571	"batch16/28c1.224.bcd"
16	174	P04572	"batch16/28c1.225.bcd"
16	175	P04573	"batch16/28c1.226.bcd"
16	176	P04574	"batch16/28c1.227.bcd"
16	177	P04575	"batch17/28c1.232.bcd"
16	178	P04576	"batch16/28c1.228.bcd"
16	179	P04577	"batch16/28c1.229.bcd"
16	180	P04578	"batch17/28c1.230.bcd"
16	181	P04579	"batch17/28c1.231.bcd"
17	100	P04497	"batch23/28c1.307.bcd"
17	103	P04500	"batch26/28c1.312.bcd"
17	113	P04506	"batch23/28c1.305.bcd"
17	114	P04507	"batch23/28c1.303.bcd"
17	115	P04508	"batch23/28c1.306.bcd"
17	124	P02739	"batch23/28c1.304.bcd"
17	126	P02741	"batch24/28c1.308.bcd"
17	129	P02744	"batch24/28c1.310.bcd"
17	130	P02745	"batch24/28c1.311.bcd"
17	132	P02747	"batch18/28c1.243.bcd"
17	136	P02748	"batch18/28c1.244.bcd"
17	137	P02749	"batch18/28c1.245.bcd"
17	138	P02750	"batch18/28c1.246.bcd"
17	139	P02751	"batch18/28c1.247.bcd"
17	140	P02752	"batch18/28c1.248.bcd"
17	141	P02753	"batch18/28c1.249.bcd"
17	142	P02754	"batch18/28c1.250.bcd"
17	143	P02755	"batch18/28c1.251.bcd"
17	144	P02756	"batch18/28c1.252.bcd"
17	145	P02757	"batch13/28c1.196.bcd"
17	146	P02758	"batch13/28c1.197.bcd"
17	147	P04611	"batch13/28c1.198.bcd"
17	148	P02759	"batch13/28c1.199.bcd"
17	149	P02760	"batch14/28c1.200.bcd"
17	150	P02761	"batch14/28c1.201.bcd"
17	151	P02762	"batch14/28c1.202.bcd"
17	152	P04580	"batch22/28c1.293.bcd"
17	153	P04581	"batch22/28c1.294.bcd"
17	154	P04582	"batch22/28c1.295.bcd"
17	155	P04583	"batch22/28c1.296.bcd"
17	156	P04584	"batch22/28c1.297.bcd"
17	157	P04585	"batch22/28c1.298.bcd"
17	158	P04586	"batch22/28c1.299.bcd"
17	159	P04587	"batch22/28c1.300.bcd"
17	160	P04588	"batch22/28c1.301.bcd"
17	161	P04589	"batch22/28c1.302.bcd"
17	162	P04590	"batch9/28c1.76.bcd"
17	163	P04591	"batch9/28c1.77.bcd"
17	163	P04591	"batch29/28c1.77r1.bcd"
17	164	P04615	"batch9/28c1.78.bcd"
17	165	P04592	"batch9/28c1.79.bcd"
17	166	P04593	"batch9/28c1.80.bcd"
17	167	P04594	"batch8/28c1.81.bcd"
17	168	P04595	"batch8/28c1.82.bcd"
17	169	P04596	"batch8/28c1.83.bcd"
17	170	P04597	"batch8/28c1.84.bcd"
17	171	P04598	"batch8/28c1.85.bcd"

17	171	P04598	"batch29/28c1.85r1.bcd"
17	172	P04599	"batch15/28c1.213.bcd"
17	173	P04600	"batch15/28c1.214.bcd"
17	174	P04601	"batch15/28c1.215.bcd"
17	175	P04602	"batch15/28c1.216.bcd"
17	176	P04603	"batch15/28c1.217.bcd"
17	177	P04604	"batch15/28c1.218.bcd"
17	178	P04605	"batch15/28c1.219.bcd"
17	179	P04606	"batch15/28c1.220.bcd"
17	180	P04607	"batch16/28c1.221.bcd"
17	181	P04608	"batch16/28c1.222.bcd"

**filenamecrj.txt**

15	92	P04429	e00401003422253b.tap
15	93	P04430	e00401003426416a.tap
15	94	P04431	e004010034260270.tap
15	95	P04432	e004010034260727.tap
15	96	P04433	e004010034261432.tap
15	97	P04434	e004010034264ba2.tap
15	98	P04435	e00401003425f8f4.tap
15	99	P04436	e00401003426515a.tap
15	100	P04437	e00401003426343c.tap
15	101	P04438	e004010034261fd6.tap
15	102	P04439	e00401003426327f.tap
15	103	P04440	e00401003425e663.tap
15	104	P04441	e004010034261aac.tap
15	105	P04442	e004010034222ea9.tap
15	106	P04443	e004010034262796.tap
15	107	P04444	e00401003425e7b9.tap
15	108	P04445	e004010034260ba5.tap
15	109	P04446	e00401003425f117.tap
15	110	P04447	e00401003425f216.tap
15	111	P04448	e004010034263cb4.tap
15	112	P04449	e00401003425ed70.tap
15	113	P04450	e00401003425ebb8.tap
15	114	P04451	e00401003425e0bd.tap
15	115	P04452	e00401003425f31e.tap
15	116	P04453	e00401003426311c.tap
15	117	P04454	e00401003425ef61.tap
15	118	P04455	e00401003425ded2.tap
15	119	P04456	e004010034262360.tap
15	120	P04457	e004010034264e4a.tap
15	121	P04458	e00401003425ee02.tap
17	92	P04489	e0040100342601d1.tap
17	93	P04490	e004010034265147.tap
17	94	P04491	e004010034263045.tap
17	95	P04492	e004010034260917.tap
17	96	P04493	e00401003425dbb6.tap
17	97	P04494	e00401003425f83b.tap
17	98	P04495	e004010034263355.tap
17	99	P04496	e00401003425e810.tap
17	101	P04498	e00401003426134d.tap
17	102	P04499	e0040100342623e7.tap
17	108	P04501	e0040100342634bf.tap
17	109	P04502	e00401003425da2f.tap
17	110	P04503	e00401003425dc22.tap
17	111	P04504	e00401003425e3a3.tap
17	112	P04505	e004010034263d87.tap
17	113	P04506	e0040100342650e5.tap
17	114	P04507	e00401003425f561.tap
17	115	P04508	e004010034263273.tap
17	116	P04509	e004010034261e40.tap
17	121	P04510	e00401003426483b.tap
17	122	P02737	e004010034263a43.tap
17	123	P02738	e004010034223b5a.tap
17	124	P02739	e00401003425fcb4.tap
17	125	P02740	e00401003426273f.tap
17	127	P02742	e0040100342622c2.tap
17	128	P02743	e00401003425ec73.tap

17	129	P02744	e0040100342650dc.tap
17	130	P02745	e00401003425fdf8.tap
17	131	P02746	e004010034262fb3.tap

**filenamecrk.txt**

15	93	P04430	22	"-03-2016/H012691645.PO"
15	102	P04439	29	"_25_16/ASLEEP-3--PACKAGE-C.P0"
15	104	P04441	29	"_25_16/ASLEEP-3--PACKAGE-C.P0"
15	105	P04442	29	"_25_16/ASLEEP-3--PACKAGE-C.P0"
15	107	P04444	22	"-03-2016/H012691646.PO"
15	109	P04446	29	"_25_16/ASLEEP-3--PACKAGE-C.P0"
15	110	P04447	29	"_25_16/ASLEEP-3--PACKAGE-C.P0"
15	114	P04451	17	"_25_16/UNKNOWN.P0"
15	118	P04455	22	"-03-2016/H012691647.PO"
17	98	P04495	17	"_25_16/UNKNOWN.P0"
17	103	P04500	29	"_25_16/ASLEEP-5--PACKAGE-E.P0"
17	110	P04503	29	"_25_16/ASLEEP-5--PACKAGE-E.P0"
17	113	P04506	29	"_25_16/ASLEEP-5--PACKAGE-E.P0"
17	115	P04508	29	"_25_16/ASLEEP-5--PACKAGE-E.P0"
17	124	P02739	22	"-03-2016/H012691644.PO"
17	129	P02744	29	"_25_16/ASLEEP-5--PACKAGE-E.P0"

**filenamecrk2.txt**

15	136	P01232	34	"798__4200201__UNKNOWN__H012873798"
17	108	P04501	34	"799__4200201__UNKNOWN__H012873799"
17	171	P04598	34	"800__4200201__UNKNOWN__H012873800"