```
 1  #!/usr/bin/perl -w
 2  # System Format = Win32
 3  #
 4  ##################################################################
 5  ##  Technical Services Scripty Thing
 6  ##  ============================
 7  ##  Author:  Xxx Xxxxxx - Technical Support Officer
 8  ##  Creation Date:  Friday, 4th August 2000.
 9  ##
10  ##  This script is designed to better manage MARC records that
11  ##  need to be sent to customers and a summary file to NLA.
12  ##  Archive attributes will play a large role in the script,
13  ##  in the future, it is hoped that the ability to automatically
14  ##  send files via FTP to NLA and the DA FTP server for customers.
15  ##
16  ##################################################################
17
18  # Variable initialise
19  $InputDIR1 = "./wuexport/";
20  $InputDIR2 = "./cuexport/";
21  $OutputDIR = "./";
22  $LastOut   = 0;
23
24  # Find the last entrman mkdiry number for output dir
25  $LastOut = &FindLastOut;
26  $NextOut = $LastOut + 1;
27  $NextOut = &GetPadString($NextOut);
28
29  # Find the files that need to be outputted in each dir
30  &GrabFileList;
31
32  &CreateInput(1);
33  &CreateInput(2);
34
35  &CopyFiles;
36
37  # &WriteOutput;
38
39
40  sub CopyFiles {
41      @CopyList = @FileList1;
42      foreach $CopyList (@CopyList) {
43             next if ($CopyList =~ /^\./);
44             next if !($CopyList =~ (/[0-9]/));
45             next if !($CopyList =~ (/txt/));
46     $x1 = $InputDIR1.$CopyList;
47     $x2 = $InputDIR1.$NextOut."/".$CopyList;
48     $Tmp = `cp $x1 $x2`;
49     print $x1,"\-\>",$x2,"\n";
50      }
51      @CopyList = @FileList2;
52      foreach $CopyList (@CopyList) {
53             next if ($CopyList =~ /^\./);
54             next if !($CopyList =~ (/[0-9]/));
55             next if !($CopyList =~ (/txt/));
56     $x1 = $InputDIR1.$CopyList;
57     $x2 = $InputDIR1.$NextOut."/".$CopyList;
58     $Tmp = `cp $x1 $x2`;
59     print $x1,"\-\>",$x2,"\n";
60      }
61  }
62
63  sub CreateInput {
64      my $What = shift;
65      if ($What == 1) {
66     print "Creating DIR: ",$InputDIR1.$NextOut,"\n";
```

```
       67         $I = mkdir ($InputDIR1.$NextOut,0777);
       68          } else {
       69         print "Creating DIR: ",$InputDIR2.$NextOut,"\n";
       70         $I = mkdir ($InputDIR2.$NextOut,0777);
       71          }
       72          if ($I) {
       73              print "Success! \n";
       74          } else {
       75              print "Fail! : $! \n";
       76          }
       77      }
       78
       79      #sub WriteOutput {
       80      #    open (OUTPUT,">$NextOut\.txt") or die ("ERROR opening output file
: $!");
       81      #    print OUTPUT "$InputDIR1\n\n";
       82      #    print @FileList1[0];
       83      #    foreach $FileList1 (@FileList1){
       84      #  print "\"",$FileList1,"\"";
       85      #          next if ($FileList1 =~ /^\./);
       86      #          next if !($FileList1 =~ (/[0-9]/));
       87      #  print OUTPUT "$FileList1\n";
       88      #    }
       89      #    print OUTPUT "$InputDIR2\n\n";
       90      #    foreach $FileList2 (@FileList2){
       91      #          next if ($FileList2 =~ /^\./);
       92      #          next if !($FileList2 =~ (/[0-9]/));
       93      #  print OUTPUT "$FileList2\n";
       94      #    }
       95      #}
       96
       97      sub FindLastOut {
       98         opendir (FINDLASTOUT_OUT,$OutputDIR);
       99         my @Files = readdir (FINDLASTOUT_OUT);
      100         closedir (FINDLASTOUT_OUT);
      101         my $Highest = $LastOut;
      102         foreach $File (@Files) {
      103                 next if ($File =~ /^\./);
      104                 next if !($File =~ (/[0-9]/));
      105                 if (int(substr($File,0,5)) >= $Highest) {
      106                         $Highest = int(substr($File,0,5));
      107                 }
      108         }
      109         print "Last Entry = ",$Highest,"\n";
      110         return $Highest;
      111      }
      112
      113      sub GrabFileList {
      114         opendir (FILELISTDIR1,$InputDIR1);
      115         @FileList1 = readdir (FILELISTDIR1);
      116         closedir (FILELISTDIR1);
      117         opendir (FILELISTDIR2,$InputDIR2);
      118         @FileList2 = readdir (FILELISTDIR2);
      119         closedir (FILELISTDIR2);
      120      }
      121
      122      sub GetPadString {
      123         my $Integer = shift;
      124         if ($Integer < 10) {
      125                 return "0000".$Integer;
      126         } elsif (($Integer >= 10) and ($Integer < 100)) {
      127                 return "000".$Integer;
      128         } elsif (($Integer >= 100) and ($Integer < 1000)) {
      129                 return "00".$Integer;
      130         } elsif (($Integer >= 1000) and ($Integer < 10000)) {
      131                 return "0".$Integer;
```

```
132         } elsif ($Integer >= 1000) {
133                 return $Integer;
134         }
135     }
136
137
138
```