```
% readSMP.m
% HPM 03/24/04 email:marshalh@colorado.edu
% this MATLAB function reads the binary SMP data
%  this was adapted from the IDL code "pntvar400.pro"
% Works only for version 302 SMP data
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% EXAMPLE:
% >>filename='D:\AVY\SMP_DATA\JOCH031804\040318_Vfeld_SMPRadarNIR\FILE0010.pnt';
% >>d=readSMP(filename)
% Note: this extracts both a force vector and temperature vector,
%  but not a depth vector to save disk space.  The respective depth
%  vectors can be created with the following commands:
% >>depth_F=d.dzF:d.dzF:(d.dzF*length(d.force));
% >>depth_T=d.dzT:d.dzT:(d.dzT*length(d.temp));
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% INPUT: filename - filename,path to SMP binary (.pnt) data file
% OUTPUT: d - structure array containing all data + header info
%         d.force = [N] vector of force measurements
%         d.dzF = [mm] distance between force samples
%         d.temp = [deg C] vector of temperature measurements
%         d.dzT = [mm] distance between temperature samples
%         d.vers = SMP version number
%         d.cF = [N/V] conversion factor to force
%         d.cP = [MPa/V] conversion factor to pressure
%         d.zero_off = [mm] zero-offset to start
%         d.year = year of measurement
%         d.month = month of measurement
%         d.day = day of measurement
%         d.hr = hour of measurement
%         d.min = minute of measurement
%         d.sec = second of measurement
%         d.xcoor = x-coordinate (if GPS on)
%         d.ycoor = y-coordinate
%         d.zcoor = z-coordinate
%         d.batt_V = battery voltage
%         d.vel = [mm/s] = average speed
%         d.fsamp = number of force samples
%         d.tsamp = number of temperature samples

function d = readSMP(filename)

%filename='D:\AVY\SMP_DATA\JOCH031804\040318_Vfeld_SMPRadarNIR\FILE0010.pnt';

fid=fopen(filename,'r','b');
d.vers=fread(fid,1,'short'); % version
d.nsamp=fread(fid,1,'long'); % number of samples
d.dzF=fread(fid,1,'float'); % distance between force samples [mm]
d.dzT=d.dzF*128; % distance between temperature samples [mm]
d.cF=fread(fid,1,'float'); % conversion factor to force (integer to [N])
d.cP=fread(fid,1,'float'); % conversion factor to pressure (integer to [MPa])
d.zero_off=fread(fid,1,'short'); % zero offset to start
```

```
d.year=fread(fid,1,'short'); % Time of measurement
d.month=fread(fid,1,'short'); %
d.day=fread(fid,1,'short'); %
d.hr=fread(fid,1,'short'); %
d.min=fread(fid,1,'short'); %
d.sec=fread(fid,1,'short'); %
d.xcoor=fread(fid,1,'double'); % GPS coordinates
d.ycoor=fread(fid,1,'double'); %
d.zcoor=fread(fid,1,'double'); %
d.batt_V=fread(fid,1,'double'); % Battery voltage
d.vel=fread(fid,1,'float'); % average speed [mm/s]

% now find the number of recorded samples
status=fseek(fid,358,'bof'); % reposition pointer
d.fsamp=fread(fid,1,'long'); % number of force samples
d.tsamp=fread(fid,1,'long'); % number of temperature samples

f_block=floor(d.fsamp/256)+1; % number of "256 byte blocks"
t_block=floor(d.tsamp/256)+1; % number of "256 byte blocks"

% now read the data:
status=fseek(fid,512,'bof'); % reposition pointer to start of data
buf=fread(fid,f_block*256+t_block*256,'short'); % total number of data points to read↙
(short integers)
d.force=buf(1:d.fsamp); % force data
temp=buf(f_block*256:f_block*256+d.tsamp); % raw temperature data
temp=temp(1:length(temp)-1);

% now convert the temperature sensor reading -- not making sense, maybe
% sensor is not working.......NOPE, JUST WRONG CALIBRATION CONSTANTS!
a_t= -0.018205007;
b_t= 0.0078710989;
c_t= -0.00098275584;
d_t= 4.2608056e-5;

i=find(temp == 0);
temp(i)=NaN;

logRt=log(temp/0.1/8.0/5.7);
d.temp=1./(a_t + b_t*logRt + c_t*logRt.^2 + d_t*logRt.^3) - 273.15 - 1.9; % temp in↙
1/100 deg C
d.temp=d.temp/100; % temp in [deg C]
d.force=d.force*d.cF; % force in [N]
%d.zF=(0:d.dzF:(d.fsamp-1)*d.dzF)'; % depth scale for force
%d.zT=(0:d.dzT:(d.tsamp-1)*d.dzT)'; % depth scale for force
d=orderfields(d,[21 3 22 4 1 2 5:20]);
d=rmfield(d,'nsamp');
status=fclose(fid);
```