# 10 MEMORY and DATA MANAGEMENT – Code Snips

**Note: These Code Snips are taken straight from the book chapter; i.e. the “Program Examples”. In some cases therefore they are not complete programs.**

/\* Program Example 10.1: Pointers example for an array average function

\*/

#include "mbed.h"

char data[]={5,7,5,8,9,1,7,8,2,5,1,4,6,2,1,4,3,8,7,9}; //define some input data

char \*dataptr; // define a pointer for the input data

float average; // floating point average variable

float CalculateAverage(char \*ptr, char size); // function prototype

int main() {

dataptr=&data[0]; // point pointer to address of the first array element

average = CalculateAverage(dataptr, sizeof(data)); // call function

printf("\n\rdata = ");

for (char i=0; i<sizeof(data); i++) { // loop for each data value

printf("%d ",data[i]); // display all the data values

}

printf("\n\raverage = %.3f",average); // display average value

}

// CalculateAverage function definition and code

float CalculateAverage(char \*ptr, char size) {

int sum=0; // define variable for calculating the sum of the data

float mean; // define variable for floating point mean value

for (char i=0; i<size; i++) {

sum=sum + \*(ptr+i); // add all data elements together

}

mean=(float)sum/size; // divide by size and cast to floating point

return mean;

}

Program Example 10.1: Averaging function using pointers

/\* Program Example 10.2: read and write char data bytes

\*/

#include "mbed.h"

LocalFileSystem local("local"); // define local file system

int write\_var, read\_var; // create data variables

int main (){

FILE\* File1 = fopen("/local/datafile.txt","w"); // open file

write\_var=0x23; // example data

fputc(write\_var, File1); // put char (data value) into file

fclose(File1); // close file

File1 = fopen ("/local/datafile.txt","r"); // open file for reading

read\_var = fgetc(File1); // read first data value

fclose(File1); // close file

printf("input value = %i \n",read\_var); // display read data value

}

****Program Example 10.2: Saving data to a file****

/\* Program Example 10.3: Read and write text string data

\*/

#include "mbed.h"

LocalFileSystem local("local"); // define local file system

char write\_string[64]; // character array up to 64 chars

char read\_string[64]; // character array up to 64 chars)

int main (){

FILE\* File1 = fopen("/local/textfile.txt","w"); // open file access

fputs("lots and lots of words and letters", File1);// put text into file

fclose(File1); // close file

File1 = fopen ("/local/textfile.txt","r"); // open file for reading

fgets(read\_string,256,File1); // read 256 chars of data

fclose(File1); // close file

printf("text data: %s \n",read\_string); // display read data string

}

****Program Example 10.3: Saving a string to a file****

/\* Program Example 10.4: Interrupt toggle switch with formatted data logging to text file

\*/

#include "mbed.h"

InterruptIn button(p30); // Interrupt on digital input p30

DigitalOut led1(LED1); // digital out to onboard LED1

Timer debounce; // define debounce timer

LocalFileSystem local("local"); // define local file system

void toggle(void); // function prototype

int main() {

debounce.start(); // start debounce timer

button.rise(&toggle); // attach the toggle function to the rising edge

}

void toggle() { // perform toggle if debounce time has elapsed

if (debounce.read\_ms()>200)

led1=!led1; // toggle LED

FILE\* Logfile = fopen ("/local/log.txt","a"); // open file for appending

fprintf(Logfile,"time=%.3fs: setting led=%d\n\r",debounce.read(),led1.read());

fclose(Logfile); // close file

debounce.reset(); // reset debounce timer

}

}

Program Example 10.4: Pushbutton LED toggle with formatted data logging

/\* Program Example 10.5: writing data to an SD card

\*/

#include "mbed.h"

#include "SDFileSystem.h"

SDFileSystem sd(p5, p6, p7, p8, "sd"); // MOSI, MISO, SCLK, CS

int main() {

FILE \*File = fopen("/sd/sdfile.txt", "w"); // open file

if(File == NULL) { // check for file pointer

printf("Could not open file for write\n"); // error if no pointer

}

else{

printf("SD card file successfully opened\n"); // if pointer ok

}

fprintf(File, "Here's some sample text on the SD card"); // write data

fclose(File); // close file

}

****Program Example 10.5: Writing data to an SD card****

/\* Program Example 10.5: writing data to an USB flash storage device

\*/

#include "mbed.h"

#include "USBHostMSD.h"

int main() {

USBHostMSD usb("usb"); // define USBHostMSD object

  while(!usb.connect()) { // try to connect a USB storage before continuing

   wait(0.5);

printf("Connecting to USB MSD\n");

  }

FILE \*File = fopen("/usb/usbfile.txt", "w"); // open file

if(File == NULL) { // check for file pointer

printf("Could not open file for write\n"); // error if no pointer

}

else{

printf("USB card file successfully opened\n"); // if pointer ok

}

fprintf(File, "Here's some sample text on the USB card"); // write data

fclose(File); // close file

}

Program Example 10.6: writing data to an USB flash storage device