Solutions - Homework 4

(Due date: Nov. 14th)

PROBLEM 1 (30 PTS)

- Refer to Activity 1 in the *High-Performance Embedded Programming with the Intel® Atom™ platform → Tutorial 8*

- First Application (Setup, then catch a `SIGINT` signal): Execute the application on the Terasic DE2i-150 Development Kit. Provide a screenshot of the execution in the Terminal. (10 pts)
  * Embed the image in your Homework 4 document.

  ✓ Based on the completion of this first application, answer the following questions (6 pts):

    ▫ What is the purpose of the line `signal (SIGINT, sig_handler)`? Mark the correct answer:
      a) Generate a signal of type `SIGINT` that will execute the function `sig_handler`.

      ~~b)~~ Configure the signal `SIGINT` so that when received, it executes the function `sig_handler`.

    ▫ If the user enters *Ctrl-c*, what signal does it generate?   ~~SIGINT~~   SIGALRM   SIGQUIT

    ▫ If the user enters *Ctrl-*, what signal does it generate?   SIGINT   SIGALRM   ~~SIGQUIT~~

- Second Application (Setup, then catch a `SIGALRM` signal): Execute the application on the Terasic DE2i-150 Development Kit. Provide a screenshot of the execution in the Terminal. (10 pts)
  * Embed the image in your Homework 4 document.
  ✓ Based on the completion of this second application, answer the following questions (4 pts):

    ▫ What is the purpose of the line `signal (SIGALRM, sig_handler)`? Mark the correct answer:
      a) Generate a signal of type `SIGALRM` that will execute the function `sig_handler`.

      ~~b)~~ Configure the signal `SIGALRM` so that when received, it executes the function `sig_handler`.

    ▫ If the user enters *Ctrl-c*, what happens?
      a) A signal of type `SIGINT` is issued that executes the function `sig_handler`.

      ~~b)~~ A signal of type `SIGINT` is issued, and it causes to exit the program.

PROBLEM 2 (10 PTS)

- Given the following code snippet:

```c
#include<stdio.h>
#include<unistd.h>
#include<signal.h>

void sig_handler(int signum){
  printf("Inside handler function\n");
}

int main(){
  int i;
  signal(SIGALRM,sig_handler);
  alarm(4);  // Scheduled alarm after 4 seconds
  alarm(1);  // Scheduled alarm after 1 seconds

  for(i=1;;i++){
    printf("%d : Inside main function\n",i);
    sleep(1);
  }
return 0;
}
```

  ✓ Answer whether the following statements are True or False:
    ▫ Two alarms will be issued: one in 1 second, and the other 3 seconds after the first.   (T)   ~~(F)~~
    ▫ An alarm will be issued in 4 seconds.   (T)   ~~(F)~~
    ▫ An alarm will be issued in 1 second.   ~~(T)~~   (F)

PROBLEM 3 (10 PTS)

- Given the following code snippet:

```c
#include<stdio.h>
#include<unistd.h>
#include<signal.h>

void sig_handler(int signum){
  printf("Inside handler function\n");
}

int main() {
  int i;
  signal(SIGALRM,sig_handler);
  alarm(2);
  alarm(0);

  for(int i=1;;i++) {
    printf("%d : Inside main function\n",i);
    sleep(1); }
  return 0;
}
```

  ✓ Answer whether the following statements are true or false:
    □ An alarm will be issued in 2 seconds.   (T)   ~~(F)~~
    □ An alarm will never be issued.   ~~(T)~~   (F)

PROBLEM 4 (10 PTS)

- Given the following code snippet:

```c
#include<stdio.h>
#include<unistd.h>
#include<signal.h>

void sig_handler(int signum) {
  printf("Inside handler function\n");
  alarm(2);
}

int main(){
  signal(SIGALRM,sig_handler);
  alarm(2);

  for(int i=1;;i++) {
    printf("%d : Inside main function\n",i);
    sleep(1); }
  return 0;
}
```

  ✓ Answer whether the following statements are true or false:
    □ An alarm is issued (and the `sig_handler` executed) every 2 seconds.   ~~(T)~~   (F)
    □ A 2-second alarm is only issued once.   (T)   ~~(F)~~

PROBLEM 5 (20 PTS)

- Refer to Activity 2 in the *High-Performance Embedded Programming with the Intel® Atom™ platform → Tutorial 8*
  ✓ Activity 2 – RTC Configuration: Execute the application on the Terasic DE2i-150 Development Kit. <u>Provide a screenshot of the execution in the Terminal</u>. You need to be `root` to execute it. (10 pts)
  * Embed the image in your Homework 4 document.

  ✓ In your own words, briefly describe the following interrupts:

Update Interrupts	Interrupts that occur when a device has its value changed. In the case of the RTC, these happen every second.
Alarm Interrupts	Interrupts that occur when the read time matches the saved time from the function call
Periodic Interrupts	Interrupts that occur at periodic intervals with programmable frequency

PROBLEM 6 (20 PTS)

▪ Attach your Project Status Report (no more than 3 pages, single-spaced, 2 columns, only one submission per group). This report should contain the current status of your project. For formatting, use the provided template (`Final Project – Report Template.docx`). The sections included in the template are the ones required in your Final Report. At this stage, you are only required to:
   ✓ You should have a very clear explanation of your application in an algorithmic fashion (i.e., like pseudo code and/or flowchart). It should be more detailed than what you presented in Homework 3 (10 pts)
   ✓ Include a much clearer parallelization strategy that you plan to apply in the components of your application. You can use pseudo-code and/or figures (10 pts)
      ▫ Identify the parallel patterns (e.g.: map, reduce, pipeline, etc.) you plan to use and where they will be applied.

▪ Only one student is needed to attach the report (make sure to indicate all the team members).