Project 3

- **Description:**
    -
        - (**30** points) Write a new system call `int get_CPU_number()` so that a process can use it to get the **number** of the CPU that executes it.

            -
                ```
                //prototype of the new system call is as follows:

                int get_CPU_number()
                ```

            - You need to write a program to prove the effectiveness of your new system call.
        - (**80** points) Write a new system call `start_to_count_number_of_process_switches()` so that a process can use it to begin to count the number of process switches the process makes. Besides, write another new system call `int stop_to_count_number_of_process_switches()` so that a process can use it to stop to count the number of process switches the process makes and return the number of process switches the process makes.

            -
                ```
                //prototype of the new system call is as follows:

                void start_to_count_number_of_process_switches()
                ```

            -
                ```
                //prototype of the new system call is as follows:

                int stop_to_count_number_of_process_switches()
                ```

        - You need to write a CPU-bound program and I/O-bound program to counter the number of process switches the programs make.
            1. What follows is a code excerpt that you need to use in your I/O-bound program.

                ```
                #define ON  1;
                #define OFF 0;

                void main()
                {                  :
                  int            a,b=0;
                  int            switch=ON;

                                 :
                  start_to_count_number_of_process_switches();
                  while(switch==ON)
                  {
                                 :
                    sleep(0.01 second);
                    printf("[%d ]",b++);

                                 :
                    if (this process has run 2 minutes)
                       switch=OFF;
                                 :
                                 :
                  }
                  a=stop_to_count_number_of_process_switches();
                  printf("\nDuring the past 2 minutes the process makes %d times process switches.\n",a);
                }
                ```

            2. What follows is a code excerpt that you need to use in your CPU-bound program.

                ```
                #define ON  1;
                #define OFF 0;

                void main()
                {                  :
                  int            a;
                  int            switch=ON;
                  float          b=0;
                                 :
                  start_to_count_number_of_process_switches();
                  while(switch==ON)
                  {
                                 :
                    b=b+1;

                                 :
                ```

```
                    if (this process has run 2 minutes)
                        switch=OFF;
                                :
                                :
        }
        a=stop_to_count_number_of_process_switches();
        printf("During the past 2 minutes the process makes %d times process switches.\n",a);
    }
```

- Hint:
  1. You can add a new field in the process descript of a process to store the number of process switches the process has made.
  2. If you want to add a new field in `struct task_struct`, append it in the end of the struct. Do NOT insert it into `struct task_struct`.
  3. Check the "Referenced Material" part of the Course web site to see how to add a new system call in Linux.
  4. You can use API `gettimeofday()` to calculate the time a process used.
  5. Process switches occur in function `__switch_to()`.
- **Project Submission:**
  - The due day of reports submission is **00:00 10th Jan. 2023**
  - The demo will be held on **10th Jan. 2023**, and **11th Jan. 2023**
  - Please fill out your available time in this form before **00:00, 7th Jan. 2023**
  - For NCU students, the demos will proceed through on-site meetings. Every member of a team must show up in the team's demo. The demo will be held in this classroom.
  - For NTHU students, the demos will proceed through google meetings. Every member of a team must show up in the team's demo. The related link of the demo is here.
  - When demonstrating your projects, the TAs will ask you some questions regarding to your projects. Part of your project grade is determined by your answers to the questions.
  - You need to submit both an electronic version and a hard-copy of your project report to the TAs.
    - The electronic versions could be sent to the TAs through new-eeclass.
    - Do not forget writing the names and student IDs of all members in your team.
    - Your report should contain:
      - Your source code
      - the execution results
  - Late submission will **NOT** be accepted.