ОС, ПОИТ-3, Лекция 07

**Computer Time**

1. **OS:** вычисление времени в компьютере

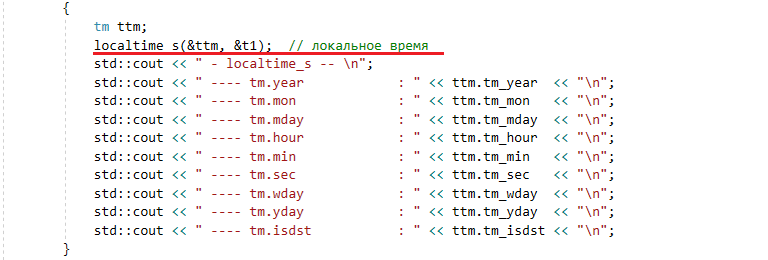
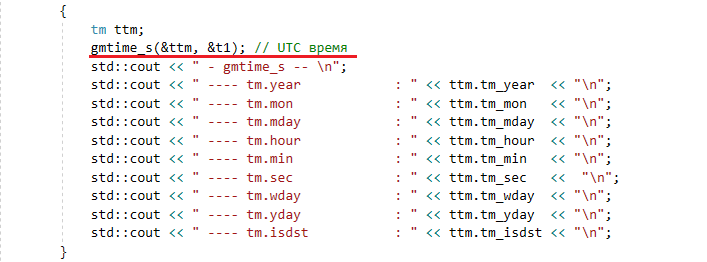
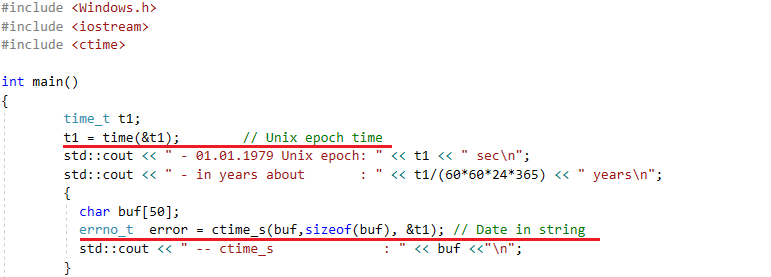


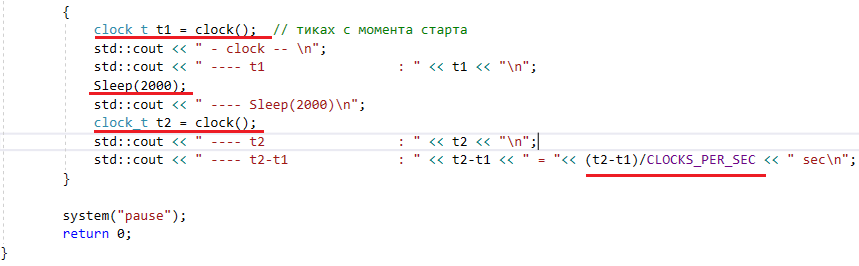




1. **OS: Социальное время не монотонное**: каждый год солнечный год увеличивается на 3 мс, люди измеряют время не точно, поэтому время от времени делались коррекции, например: в 1582 Папа Григорий XIII пропустил 10 дней календаря. Вычисление даты от Рождества по секундам не получится. Поэтому **эпоха Unix (POSIX-время)** c 01.01.1970 0:00:00 в секундах. Используется 32 бита для представления числа. В 2038 г. счетчик перейдет в область отрицательных чисел. Секунда координации (по сообщению Международной службы вращения Земли, серверы точного времени): последняя секунда 30.06 или 31.12.

1. **OS:Universal Coordinated Time (UCT):** универсальное согласованное время (на Гринвичском меридиане, раньше GMT – Greenwich Meridian Time), Международное бюро мер и весов (Париж), усредненное значение полученное на основе данных 50 лабораторий, оборудованных атомными часами (цезий-133) - TAI (International Atomic Time), расхождение с солнечными часами примерно 3мс (атомные часы отстают) в сутки, коррекция при ошибке в 800 мс.
2. **OS: Коротковолновые радиостанции с позывным WWV**: выдают сигнал в начале каждой секунды UTC с точностью до ±10мс.
3. **OS: Спутники Geostationary Environment Operational Satellite (GEOS):** выдают сигнал в начале каждой секунды UTC с точностью до ±5мс.
4. **OS:** c 01.01.1970эпоха Unix, 32 бита, в секундах, до 2038.
5. **OS:** Windows: время, дата.

****

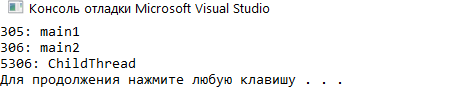
****

1. **OS:** ожидающий таймер: объект синхронизации, два состояния: сигнальное – наступление заданного момента времени; несигнальное (активное и пассивное состояние) - ждет наступления заданного момента времени.

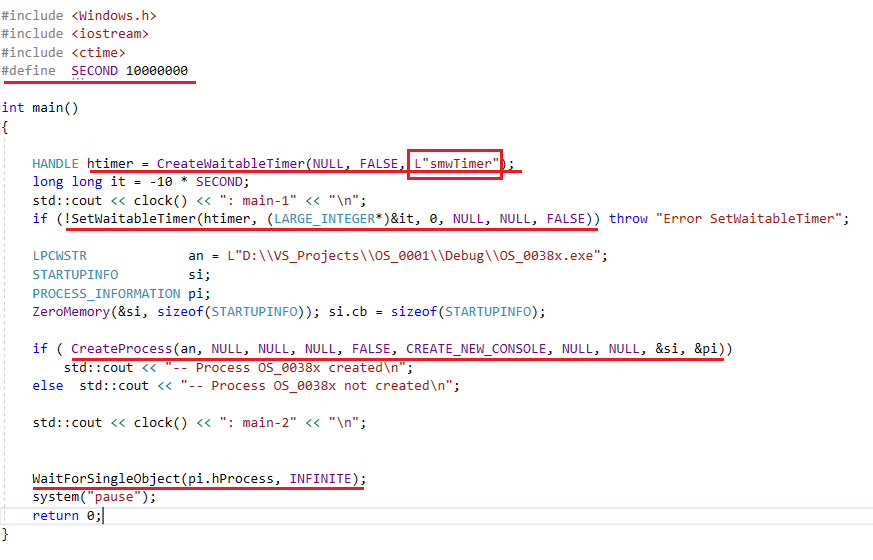


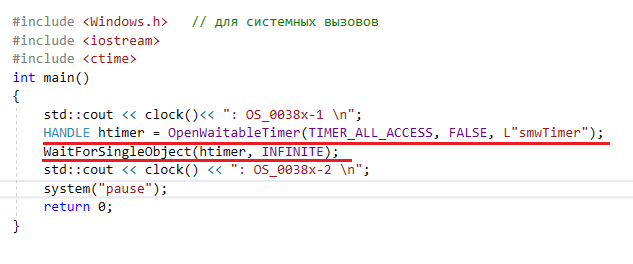
1. **OS:** Windows:CreateWaitableTimer, SetWaitableTimer, WaitForSingleObject.

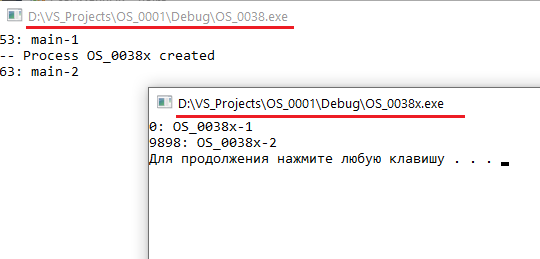
****

****

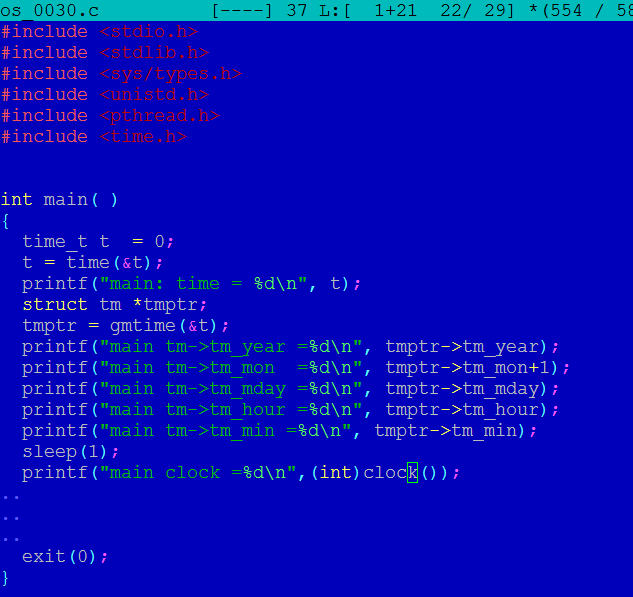
1. **OS:** Windows: OpenWriteTimer

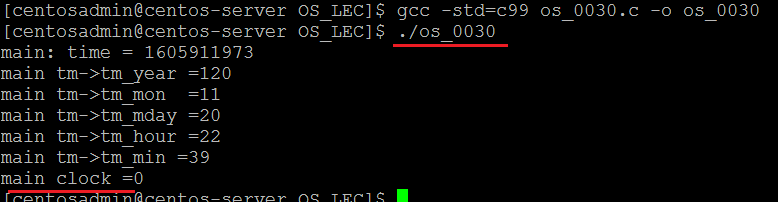
****

****

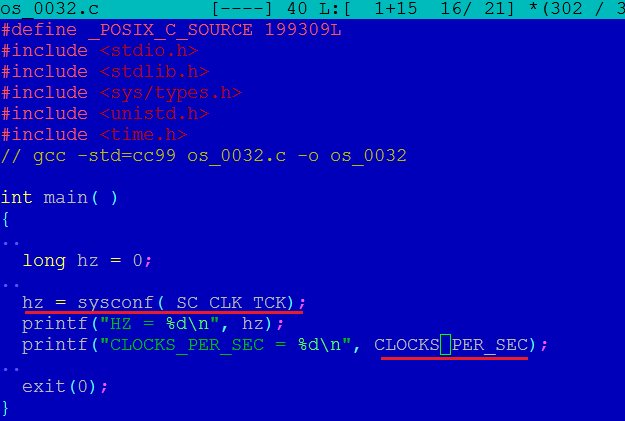
****

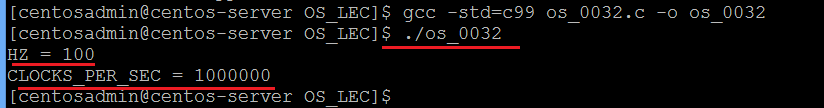
1. **OS:** Windows: CancelWriteTimer
2. **OS:** Windows: процедуры завершения ожидания
3. **OS:** Windows: Sleep()
4. **OS:** Linux: time\_t, tm, time(), ctime, ctime\_r, gmtime(), gmtime\_r, localtime(), clock(), …, POSIX, CLOCK\_PER\_SEC

****

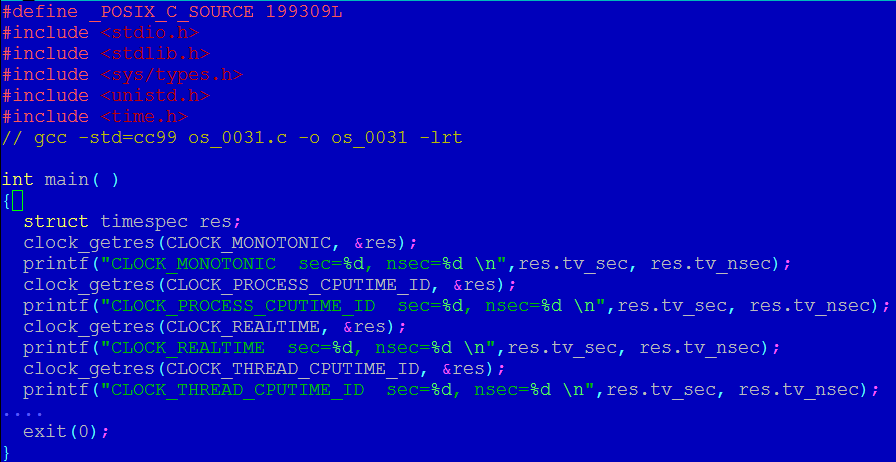
****

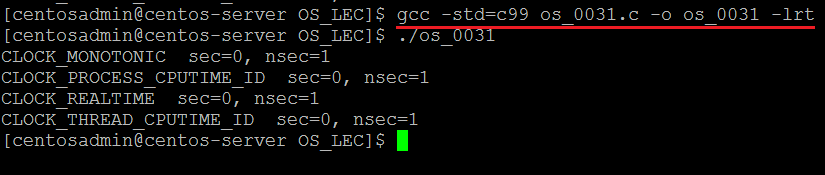
1. **OS:** Linux: четыре типа часов: REALTIME – системное время(настенное), MONOTONIC – с начала загрузки OS(монотонно возрастает), PROCESS – процессорное время( затраченное процессом), THREAD – процессорное время(затраченное потоком).
2. **OS:** Linux: относительное время (через 5 сек.), абсолютное время(29.11.2020,9:50).
3. **OS:** Linux: HZ – частота системного таймера (обычно, 100,250, **1000**), параметр ядра.

****

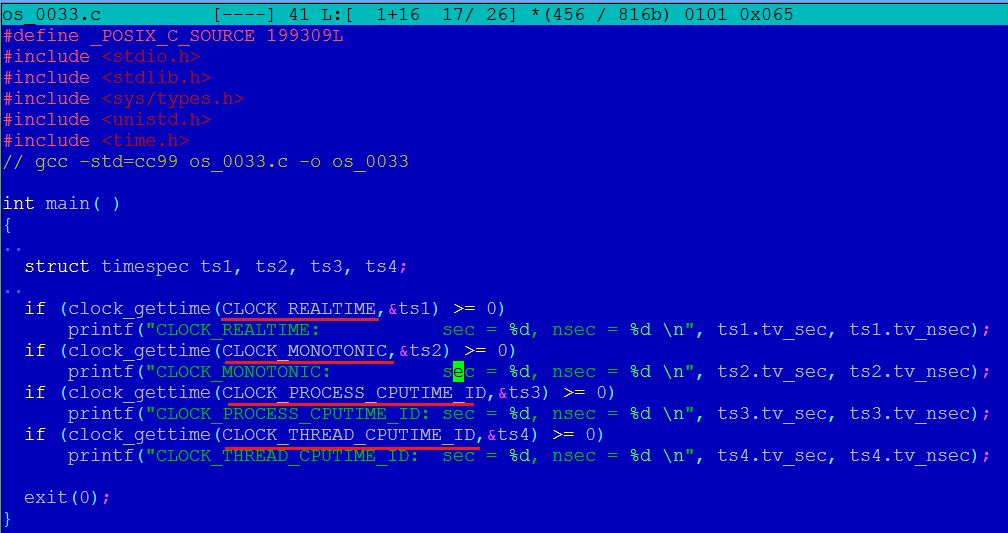
****

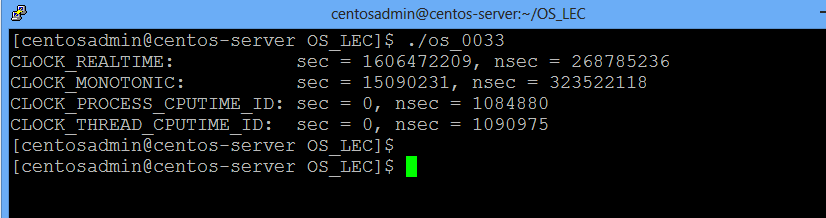
1. **OS:** Linux: точность часов

****

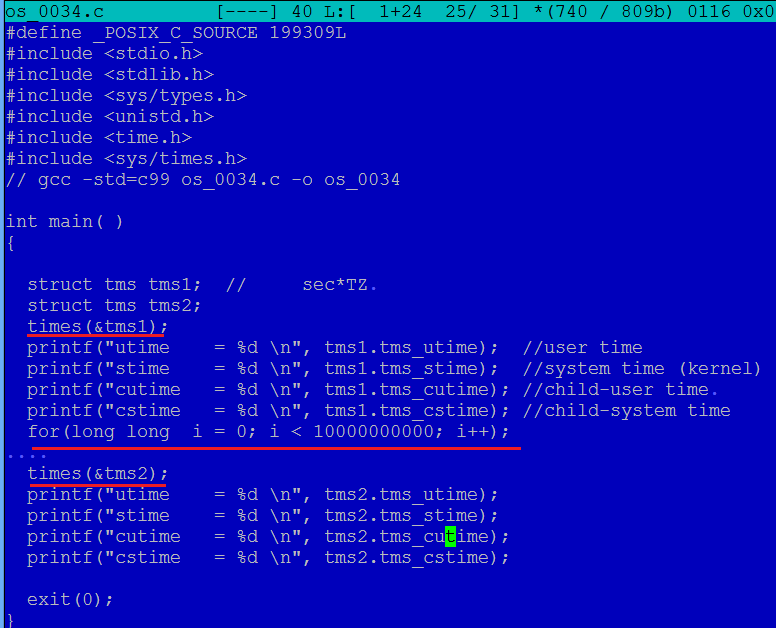
****

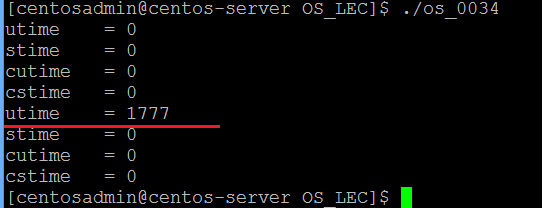
1. OS:Linux: расширенный интерфейс, timespec, clock\_gettime



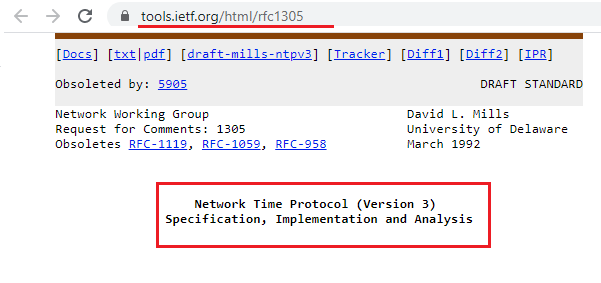


1. **OS:** Linux: измерить время: times

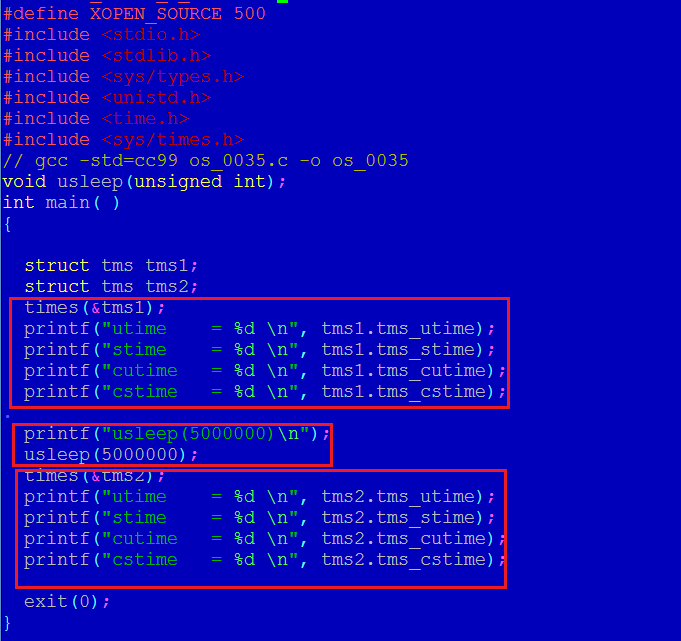
****

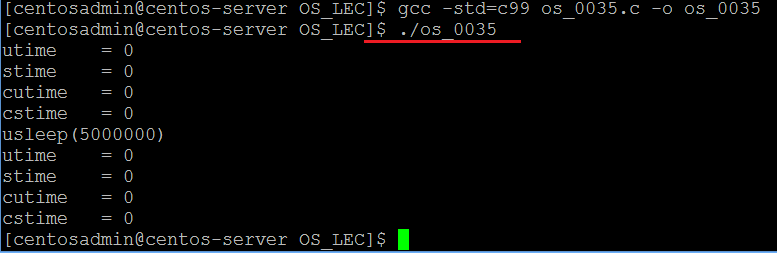
****

1. **OS:** Linux: установка точного времени: stime, settimeofday, clock\_settime.
2. **OS:** Linux: подстройка системных часов: adjtime – замедление или ускорение системных часов на заданную величину (плавная корректировка часов), adjtimex – более мощный механизм подстройки часов (RFC 1305)

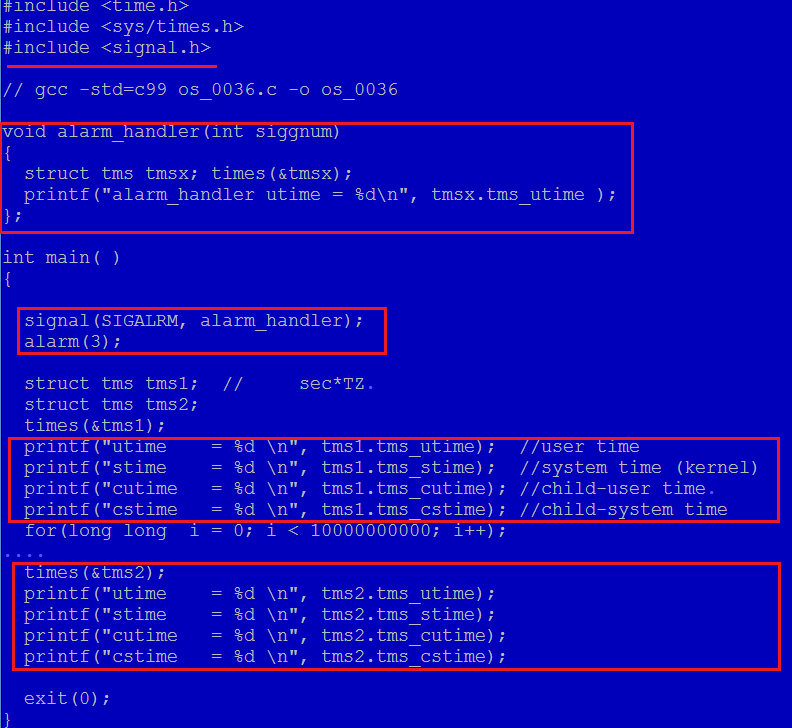


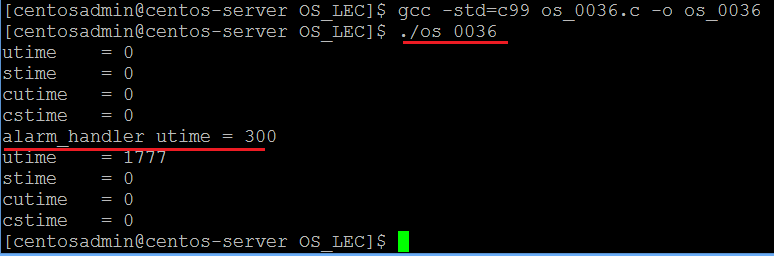
**OS:** Linux: sleep()- засыпание (сек), usleep()- засыпание (микросек = 10-6), nanosleep(10-9) – может досыпать.



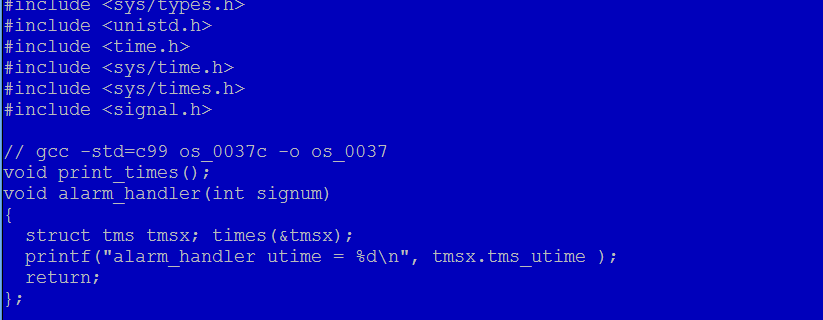


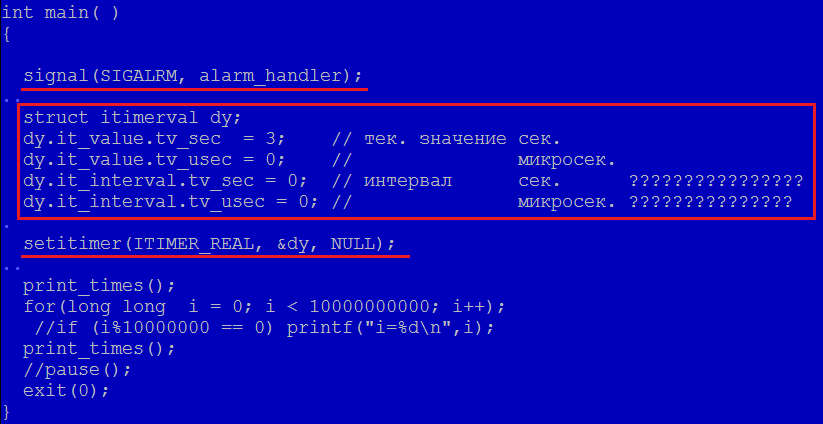
1. **OS:** Linux: alarm – послать от ядра процессу сигнал через заданный интервал времени.

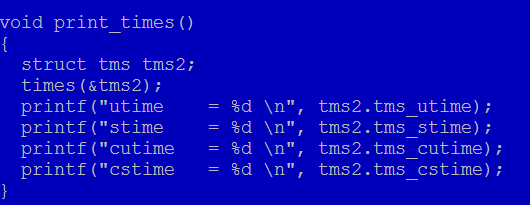
****

****

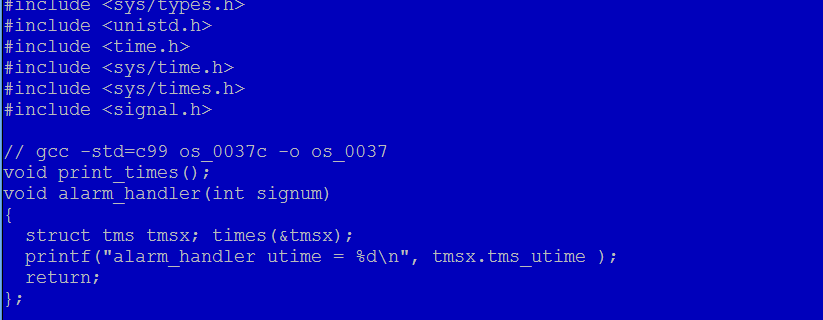
1. **OS:** Linux: интервальные таймеры setitimer. REAL – реальное время, VIRUAL – выполнение пользовательского кода(для профилирования),PROF – выполнение ядра от имени процесса (для профилирования).

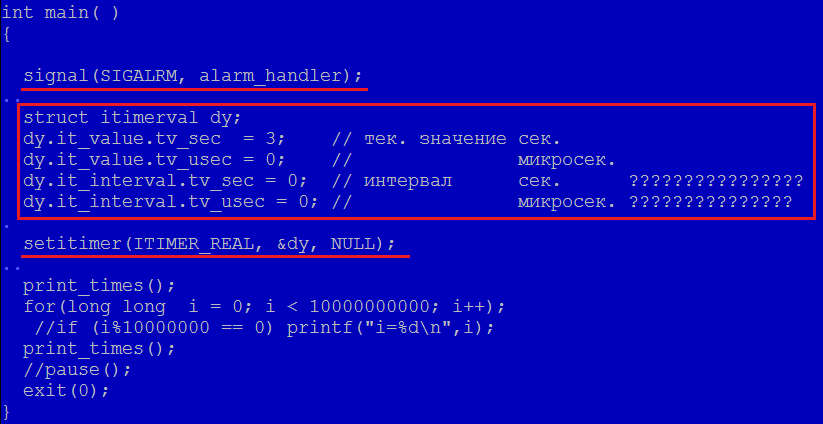
****

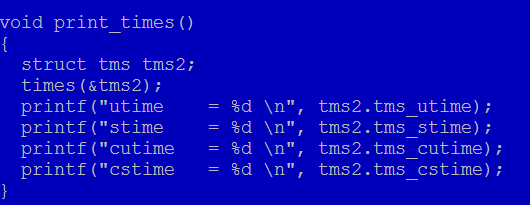
****

****

1. **OS: linux:** расширенные таймеры: timer\_create, timer\_settime, timer\_gettime(получение остатка времени), timer\_getoverrun (к-во превышений таймера), timer\_delete.

****

****

****

1. **OS:** итоги

* тактовые генераторы и специальные регистры – основа для вычисления времени;
* для хранения системного времени используется энергонезависимая память;
* аппаратное прерывание системного таймера имеет высший приоритет;
* социальное время не монотонное;
* POSIX-время, Unix epoch 01.01.1970 00:00, 32bit, sec, 2038;
* UCT – универсальное согласованное время, Международное бюро мер и весов (Париж), усредненное значение полученное на основе данных 50 лабораторий; распространители: WWV(радиостанции), GEOS-спутники; NTP-служба в Internet;
* Windows: тики, CLOCKS\_PER\_SEC = 1000.
* Windows: стандартная библиотека С: специальные типы (time\_t, tm, clock\_t), функции(time, gmtime, localtime, clock, …);
* Windows: ожидающий таймер, объект ядра, для синхронизации, состояния (активное/пассивное, сигнальное/несигнальное), периодические и одноразовые, синхронизация потоков и процессов; создать, взвести, ожидать, удалить;
* Windows: засыпание Sleep();
* Linux: стандартная библиотека С: специальные типы (time\_t, tm,… ), функции(time, gmtime, localtime, clock, …), расширенный интерфейс clock\_gettime, times;
* Linux: 4 типа часов: REALTIME – системное время(настенное), MONOTONIC – с начала загрузки OS(монотонно возрастает), PROCESS – процессорное время( затраченное процессом), THREAD – процессорное время(затраченное потоком); точность clock\_getres для каждого типа часов
* Linux: HZ – частота системного таймера (обычно, 100,250, **1000**), параметр ядра, sysconf(\_SC\_CLK\_TCK); может быть не согласовано CLOCKS\_PER\_SEC;
* Linux: установка точного времени, подстройка часов,
* Linux: засыпание: sleep(), usleep(), nanosleep();
* Linux: alarm, SIGALRM – реализация ожидающего таймера;
* Linux: интервальные таймеры setitimer. REAL – реальное время, VIRTUAL – выполнение пользовательского кода(для профилирования),PROF – выполнение ядра от имени процесса (для профилирования).