Elaborato 3: Utilizzo Thread.
Consegna: entro il 11 Luglio 2010 ore 9:00.
Modalità di consegna:

1) Rinominare il file contenente l'elaborato con il proprio numero di matricola. Si ricorda che la consegna è individuale, pertanto ogni studente dovrà consegnare una copia dell'elaborato.
2) Riportare in calce al file contenente l'elaborato un commento che includa: matricola, nome e cognome, data di consegna, titolo dell'elaborato.
3) Fare l'upload del file su http://amarena.sci.univr.it/
    a. Seguire i link: Accesso pubblico → Laboratorio Sistemi Operativi 2010 → "Nome_docente_del_corso"
    b. A questo punto dovreste trovarvi all'interno di anonymous / Laboratorio Sistemi Operativi 2010 / Nome_docente_del_corso
    c. Cliccare sulla freccia alla destra della voce Elaborato 3 (sotto la colonna Azione), quindi su Nuovo → Documento
    d. Compilare i campi del form che appare inserendo il file di cui si vuole fare l'upload in "File locale", il vostro nome, cognome e n° di matricola su "Nome del documento".
    e. Premere OK
4) Si ricorda inoltre che non si potranno né modificare né visualizzare i file di cui è stato fatto l'upload.
5) Per qualunque problema durante la sottomissione dell'elaborato contattare il docente del relativo corso (Bombieri per Informatica Multimediale, Carra per Informatica Generale).
6) Dopo la scadenza del 11/07 non sarà più possibile effettuare l'upload dell'elaborato. Chi non avrà consegnato perderà definitivamente il diritto di fare l'esame nella modalità orale.

Testo dell'elaborato

Scrivere un'applicazione C che, sfruttando le thread, implementa una (rudimentale) funzione crittografica di hash.

Si consideri come input un file di testo contenente un numero qualsiasi di caratteri. La funzione di hash lavora su blocchi da 64 caratteri: qualora il testo del file di input non contenesse un multiplo di 64 caratteri, si dovrà aggiungere un padding finale costituito da caratteri "0" (tanti quanti sono necessari per arrivare ad un multiplo di 64). Se ad esempio il testo ha 10 caratteri, la funzione di hash lavorerà su quei 10 caratteri a cui sono stati aggiunti 54 "0" consecutivi. Se il testo ha 120 caratteri, lavorerà sui 120 caratteri a cui sono stati aggiunti 8 "0" consecutivi.

Sia b[i] il blocco i-esimo di 64 caratteri, con i = 0…k, dove k e' il numero di multipli di 64 caratteri di cui è composto il testo. Ciascun blocco b[i] viene suddiviso in 4 sottoblocchi da 16 caratteri ciascuno chiamati x[i], y[i], w[i], e z[i]. Il lavoro di processing del testo viene suddiviso tra 4 thread: ciascun thread è responsabile di uno dei sottoblocchi x, y, w, e z. I thread eseguono le seguenti operazioni:

- il thread 1 prende il blocco x[0], fa lo shift a sinistra di 8 bit, e ne fa lo XOR bit a bit con il blocco x[1], ottenendo il blocco x'[1]. Ripete lo stesso procedimento usando il blocco x'[1] e x[2] e così via fino a quando ha terminato i blocchi. Si noti che se il testo ha solo 1 blocco, viene effettuata solo l'operazione di shift. Il risultato finale sarà il blocco x'[k].

- il thread 2 prende il blocco y[0], fa lo shift a destra di 8 bit, e ne fa l'AND bit a bit con il blocco y[1], ottenendo il blocco y'[1]. Ripete lo stesso procedimento usando il blocco y'[1] e y[2] e così via fino a quando ha terminato i blocchi. Si noti che se il testo ha solo 1 blocco, viene effettuata solo l'operazione di shift. Il risultato finale sarà il blocco y'[k].

- il thread 3 prende il blocco w[0], fa lo shift a sinistra di 16 bit, e ne fa l'OR bit a bit con il blocco w[1], ottenendo il blocco w'[1]. Ripete lo stesso procedimento usando il blocco w'[1] e w[2] e così via fino a quando ha terminato i blocchi. Si noti che se il testo ha solo 1 blocco, viene effettuata solo l'operazione di shift. Il risultato finale sarà il blocco w'[k].

- il thread 4 prende il blocco z[0], fa lo shift a destra di 16 bit, e ne fa l'AND bit a bit con il blocco NOT(z[1]), dove l'operazione di NOT(.) inverte il valore di ciascun bit dell'argomento, ottenendo il blocco z'[1]. Ripete lo stesso procedimento usando il blocco z'[1] e z[2] e così via fino a quando ha terminato i blocchi. Si noti che se il testo ha solo 1 blocco, viene effettuata solo l'operazione di shift. Il risultato finale sarà il blocco z'[k].

L'hash finale viene calcolato come lo XOR bit a bit tra x'[k], y'[k], w'[k] e z'[k]. Tale hash viene salvato in un file di testo che ha lo stesso nome del file di input, ma con estensione .hash.