

Corso: Sistemi Operativi (9 cfu) - Appello n. 2 - del 28 Giugno 2010	
Cognome Nome (in stampatello):	
Matricola o estremi del documento:	

N.	Testo dell'esercizio	Punti (tempo)
	<i>Soluzione dello studente o testo descrittivo</i> Nel caso non sia sufficiente lo spazio, specificare "la soluzione continua sul foglio n. X allegato"	%
1	<p>Descrivere brevemente il concetto di processo e fornire una descrizione dei passi fondamentali necessari per l'esecuzione della system call <code>fork()</code> dalla sua invocazione alla esecuzione del nuovo processo.</p>	5 (15')
2	<p>Il candidato indichi l'algoritmo di scheduling più adatto per un sistema informativo di tipo batch motivando la scelta in funzione delle caratteristiche del carico di lavoro..</p>	5 (15')
3	<p>Descrivere la struttura della tabella delle pagine invertita indicandone i vantaggi nel caso di processore a 64 bit.</p>	5 (15')

<p>N.</p> <p>4</p>	<p>Considerando un tempo di accesso alla memoria di 100 nsec e di 10 msec il tempo per la risoluzione di un page fault, se il sistema possiede un 1 Gb di memoria fisica e l'accesso a tutte le pagine è equiprobabile, quale la taglia massima di memoria virtuale allocabile per tenere il tempo medio di accesso < 200 nsec ?.</p>	<p>5 (15')</p> <p>%</p>
<p>N.</p> <p>5</p>	<p>Scrivere un programma C che crea un processo figlio che, a sua volta, crea un file con nome casuale (intero di 8 cifre) una volta al secondo (<code>sleep()</code>) scrivendovi a caso una quantità di byte casuali compresa tra 1.000 e 100.000 byte. Il processo padre, dopo aver avviato il figlio, una volta al secondo, scorrerà la stessa directory (<code>opendir()</code> e <code>readdir()</code>) e stamperà la lista dei file che vi trova ordinata per dimensione crescente. N.B. Non è necessario implementare la funzione di ordinamento, basterà invocare la funzione di libreria <code>qsort()</code>.</p>	<p>14 (60')</p> <p>%</p>