



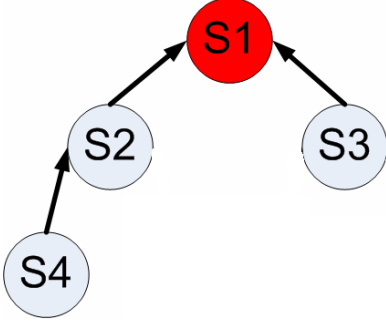
<p>نام درس: سیستم عامل پیشرفته نیم‌سال اول 87-88 استاد درس: دکتر رسول جلیلی صفحه: 1</p>	<p>باسمه تعالی تمرین سری دوم 87/08/27 موعد تحویل: 86/09/15 (تا ساعت 23:59)</p>	 <p>دانشگاه صنعتی شریف دانشکده مهندسی کامپیوتر</p>
---	--	--

<p style="text-align: right;">نکات قابل توجه:</p>	
<p>- پاسخ‌های خود را در صورت امکان به صورت الکترونیکی به آدرس Mahrooghi@ce.sharif.edu با عنوان [OS2:Assignment#2] ارسال نمایید.</p> <p>- حل تمرین‌های درس باید به صورت فردی انجام پذیرد.</p> <p>- به پاسخ‌هایی که رونوشت یکدیگر باشند، هیچ نمره‌ای تعلق نمی‌گیرد.</p> <p>- در صورت تاخیر در ارسال پاسخ، به ازای هر روز 25٪ نمره کل تمرین، کسر می‌گردد.</p>	
<p>1 n پردازش که حداکثر $(n-1)/3$ آنها خطادار است را در نظر بگیرید. فرض کنید که این n پردازش می‌خواهند روی مقداری که پردازش p_0 تعیین می‌کند (0 یا 1) توافق Byzantine کنند. فرض کنید که ارسال هر پیغام بین 0 تا 1 واحد زمانی و پردازش هر پیغام (به صورت گروهی) برای یک پردازش سالم 1 تا 2 واحد زمانی و برای یک پردازش خطادار 0 تا 1 واحد زمانی، زمان می‌برد. زمان مورد نیاز برای سایر عملیات‌ها ناچیز فرض می‌گردد.</p> <p>حداکثر و حداقل زمان مورد نیاز برای توافق از زمانی که پردازش p_0 اولین پیغام را ارسال می‌کند تا زمانی که همه پردازش‌های صحیح روی مقدار آن توافق می‌کنند، با استفاده از الگوریتم Lamport-Shostak-Pease و Delov چقدر است؟ تعداد پیغامی که هر پردازش دریافت و ارسال می‌کند چقدر است؟</p>	1
<p>2 نشان دهید که سه مساله توافق Byzantine، سازگاری محاوره‌ای (Interactive Consistency) و اجماع (Consensus) معادلند.</p>	2
<p>3 به الگوریتم زیر که برای توافق Byzantine در حضور پیغامهای امضا شده (Signed یا Authenticated) ارائه شده توجه کنید:</p> <p>Initialization: $V_i = \{ \}$</p> <p>The transmitter (node p_0) signs its value and sends to all other nodes. For each i:</p> <ul style="list-style-type: none"> ▪ If node p_i receives a message of the form $(a:0)$ from the transmitter then <ul style="list-style-type: none"> ○ It sets V_i to $\{a\}$. ○ It sends the message $(a:0,i)$ to all other nodes. ▪ If node p_i receives a message of the form $(a:0,j_1,j_2,\dots,j_k)$ and a is not in V_i, then <ul style="list-style-type: none"> ○ It adds a to V_i ○ If $k < n$, it sends the message $(a:0,j_1,j_2,\dots,j_k,i)$ to every node other than j_1,j_2,\dots,j_k <p>For each i,</p> <ul style="list-style-type: none"> ▪ When node p_i receive no more messages, it considers the final value as choice (V_i). <p>الف) نشان دهید که با استفاده از این الگوریتم، 5 پردازش با حضور 2 پردازش خطادار نیز قادر به توافق هستند.</p> <p>ب) نشان دهید که تعداد دورهای (round) مورد نیاز برای توافق n پردازش در حضور پیغامهای امضا شده و با استفاده از این الگوریتم برای m پردازش خطادار، $m+1$ است.</p> <p>ج) آیا شرط $n > m+1$ برای رسیدن به توافق الزامی است؟ پاسخ خود را اثبات کنید.</p> <p>راهنمایی: به عنوان مثال پردازش p_2 می‌تواند پیغامی به شکل $(0:2)$ را بفرستد، اما نمی‌تواند پیغام $(1:0,1)$ را به شکل $(0:0,1,2)$ ارسال نماید.</p>	3

<p>نام درس: سیستم عامل پیشرفته نیم‌سال اول 87-88 استاد درس: دکتر رسول جلیلی صفحه: 2</p>	<p>باسمه‌تعالی تمرین سری دوم 87/08/27 موعده تحویل: 86/09/15 (تا ساعت 23:59)</p>	 <p>دانشگاه صنعتی شریف دانشکده مهندسی کامپیوتر</p>
---	---	--

<p>الف) مثالی از reliable broadcast بزنید که uniform نباشد. ب) مثالی از یک reliable broadcast بزنید که FIFO باشد ولی Causal نباشد.</p>	<p>4</p>
<p>ویژگی‌های زیر را در نظر بگیرید: ○ اعتبار: اگر یک پردازش درست، پیغام m را پخش کند، آنگاه همه پردازش‌های درست در نهایت m را دریافت (deliver) خواهند کرد. ○ عدم تکرار: یک پیغام بیش از یک بار توسط یک پردازش دریافت نمی‌شود. ○ عدم ایجاد: اگر پردازش‌های m را دریافت کند، m قبلاً (توسط پردازش‌های) پخش شده است. ○ ترتیب علی: اگر پردازش‌های درست m را دریافت کند، پیش از آن تمامی m‌هایی که برای آنها رابطه $m \rightarrow m'$ برقرار است را دریافت کرده است. آیا می‌توانیم الگوریتم پخش ابداع کنیم که تمامی این ویژگیها را در بر داشته باشد، ولی ویژگی توافق (Agreement) پخش Reliable را نقض نماید؟</p>	<p>5</p>
<p>نشان دهید که الگوریتم ارائه شده در درس برای پیاده‌سازی Causal Broadcast با استفاده از FIFO Broadcast از نظر پیغامی، پیچیدگی بالایی داشته و ناکارآمد است. یک الگوریتم کارآمدتر فقط با استفاده از Reliable Broadcast ارائه دهید. راهنمایی: از vector clock برای حفظ ترتیب علی استفاده کنید.</p>	<p>6</p>
<p>نشان دهید که در الگوریتم‌های Lamport و Ricart-Agrawala، پردازش‌ها به ترتیب صعودی زمان مهرها وارد CS می‌شوند. آیا این گفته در مورد الگوریتم Maekawa نیز صحیح است؟</p>	<p>7</p>
<p>فرض کنید که از روش ساده زیر برای اعمال Mutual Exclusion استفاده می‌کنیم: سایت‌ها به صورت یک ring منطقی مرتب شده‌اند و یک مهره یکتا در درون ring در حال چرخش بوده و از سایتی به سایت بعد می‌رود. هنگامی که یک سایت نیاز به اجرای CS دارد، منتظر مهره می‌ماند و پس از به دست آوردن مهره، CS را اجرا نموده و سپس مهره را به سایت بعد می‌فرستد. اگر، سایتی در هنگام به دست آوردن مهره نیازی به اجرای CS نداشته باشد، در زمان صفر آنرا به سایت بعد می‌فرستد. کارایی، تأخیر همگامی و زمان پاسخ در هر یک از حالات زیر چقدر است؟ الف) در زمانی که بار سیستم سبک است. ب) در زمانی که بار سیستم سنگین است. فرض کنید تعداد سایت‌ها برابر N، تأخیر پیغام/مهره برابر T و زمان اجرای CS برابر E است.</p>	<p>8</p>
<p>چهار سایت s_1 تا s_4 را در نظر بگیرید. فرض کنید که از یک روش مبتنی بر مهره در برقراری Mutual Exclusion استفاده می‌کنیم. در حالت ابتدایی، مهره در اختیار سایت s_1 می‌باشد. درخواست‌ها برای ورودی به CS به شکل زیر می‌باشد: 1) سایت s_2 درخواست ورود به CS را می‌دهد. 2) سایت s_4 درخواست ورود به CS را پس از خروج s_2 از CS می‌دهد. 3) سایت s_1 درخواست ورود به CS را در حالی که s_4 در CS است، می‌دهد. الف) برای الگوریتم سوزوکی و کاسامی، مقادیر آرایه‌های LN و آرایه‌های RN_i را برای i از 1 تا 4 در طی این درخواست‌ها دنبال کنید. ب) برای الگوریتم سینگهال، مقادیر آرایه‌های TSV و TSN و آرایه‌های SV_i و SN_i را برای i از 1 تا 4 در طی این درخواست‌ها دنبال کنید.</p>	<p>9</p>

<p>نام درس: سیستم عامل پیشرفته نیم‌سال اول 87-88 استاد درس: دکتر رسول جلیلی صفحه: 3</p>	<p>باسمه تعالی تمرین سری دوم 87/08/27 موعد تحویل: 86/09/15 (تا ساعت 23:59)</p>	 <p>دانشگاه صنعتی شریف دانشکده مهندسی کامپیوتر</p>
---	--	--

<p>ج) برای الگوریتم درخت‌مبنای ریموند، نحوه تغییرات درخت و بردار request-q را با فرض اینکه، شکل اولیه درخت به صورت زیر باشد، دنبال کنید.</p>  <p>(RN_iها را می‌توانید به صورت یک آرایه RN با ابعاد 4*4 نشان دهید و به همین ترتیب S_{V_i} و S_{N_i})</p>	
<p>10 در بسیاری از مواقع، یک سایت، قبل از رسیدن درخواست از یک سایت دیگر، بارها CS را اجرا می‌کند. توضیح دهید که چرا الگوریتم Ricart-Agrawala در چنین مواردی ناکارآمد است. الگوریتم را به گونه‌ای تغییر دهید که این نقصان برطرف گردد. آیا الگوریتم جدید شرط Liveness را دارا می‌باشد؟</p>	

موفق باشید