

Fetch	Decode	Allocation	Issue	Execute	Commit
<b>4 Estágios</b> <ul style="list-style-type: none"> <li>- Branch + Fetch</li> <li>- Cálculo end.</li> <li>- Acesso a mem</li> <li>- Align</li> </ul>	<b>1 Estágio</b> <ul style="list-style-type: none"> <li>- Decodificação</li> </ul>	<b>3 Estágios</b> <ul style="list-style-type: none"> <li>- Register Rename</li> <li>- Data ready info</li> <li>- Data write</li> </ul>	<b>3 Estágios</b> <ul style="list-style-type: none"> <li>- Wakeup</li> <li>- Select</li> <li>- Drive</li> </ul>	<b>2 Estágios</b> <ul style="list-style-type: none"> <li>- Execute</li> <li>- Write regs</li> </ul>	<b>1 Estágio</b> <ul style="list-style-type: none"> <li>- Reorder e Commit</li> </ul>
<b>4 Ciclos base</b> +10 se L1i miss +50 se L2 miss +100 se L3 miss	<b>1 Ciclo base</b>	<b>1 Ciclo base</b> (2 executam junto com Issue)	<b>3 Ciclos base</b> + #Ex, se dep.	<b>2 Ciclos base</b> +10 se L1d miss +50 se L2 miss +100 se L3 miss	<b>1 Ciclo base</b>
<b>Configurações:</b> # instruções # bits gshare					
<b>Acesso a Memória</b> Caso haja acesso a memória, ciclos extras serão contados e próxima instruções também respeitaram esse ciclos.			<b>Dependências</b> Dependências de dados e disponibilidade de recursos são respeitados.	<b>Bypass</b> No ciclo em que se faz writeback, o dado está disponível para a próxima instrução. <b>Acesso a Memória</b> São ordenados, misses de escrita e leitura geram misses.	<b>Branch Prediction</b> Caso haja erro na predição, a próxima instrução executa no ciclo seguinte, ao do commit.