الاسبوع الثامن عشــر

جدولة وحدة المعالجة المركزية
CPU Scheduling

جدولة وحدة المعالجة المركزية CPU Scheduling

وحدة المعالجة المركزية (CPU) من أهم أجزاء الحاسوب والتي تقوم بتنفيذ العمليات (Processes) ولكنها تقوم بتنفيذ عملية واحدة في الوقت الواحد وهناك الكثير من العمليات التي تحتاج للتنفيذ ، فكيف يتم التنسيق والتنظيم بين العمليات ؟

إن مجدول وحدة المعالجة (CPU Scheduler) هو المسؤول عن التنسيق بين العمليات باستخدام خوارزميات مختلفة .

الغرض من جدولة العمليات هي اختيار عملية من العمليات الموجودة في الذاكرة للتنفيذ من قبل المعالج ، بحيث يكون المعالج دوما مشغولا مما يزيد من كفاءته .

حلقة العمل داخل وحدة المعالجة المركزية – وعملية الإدخال/الإخراج CPU-I/O Burst Cycle

تنفيذ العملية (Process) يتألف من حلقة من معالجة العملية داخل وحدة المعالجة المركزية وانتظار عملية الإدخال والإخراج .

كل عملية تمر بدورة متبادلة ما بين العمل داخل وحدة المعالجة المركزية (CPU burst) والعمل في وحدات الإدخال والإخراج (I/O burst) ليتم تنفيذها ،فعند طلب فتح البرنامج تبدأ العملية بالعمل داخل وحدة المعالجة المركزية (CPU burst) ثم العمل في وحدات الإدخال والإخراج (I/O burst) ، ويستمر في الحالتين بشكل متبادل ، وينتهي بالعمل داخل وحدة المعالجة المركزية (CPU burst) بطلب نظام التشغيل لإنهاء التنفيذ لسبب ما ، إما العملية انتهت أو لحدوث خطأ تسبب بإلغائها .

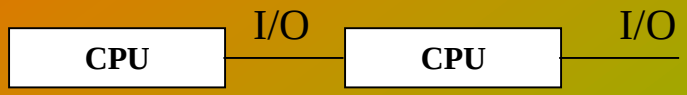
يمكن تصنيف العمليات إلى نوعين حسب العمل :

1- I/O bound processes :

العمليات من هذا النوع تقضي الوقت الكثير في الإدخال والإخراج والقليل من الوقت في العمليات الحسابية التي تحتاج وحدة المعالجة المركزية (CPU).

2- CPU bound processes:

العمليات من هذا النوع تقضي الوقت الكثير في المعالجة والقليل في طلب الإدخال والإخراج .



أنواع المجدول (Schedulers)

1- المجدول طويل الأمد (Long-term scheduler): ومختصره L.T.S

ويطلق عليه مجدول المهام أيضا (Job scheduler) ، يختار العمليات (processes) من بركة المهام (Job pool) والموجودة في القرص الصلب ويقوم بتحميلهم إلى الذاكرة ويعطيهم الموارد التي تحتاجها ما عدا المورد الثمين (وحدة المعالجة المركزية CPU) .

2- المجدول قصير الأمد (Short-term scheduler): ومختصره S.T.S

يختار عملية واحدة من بين مجموعة من العمليات الجاهزة للتنفيذ بعد إشارة المقاطعة أو بعد استدعاء النظام ويمنحها المعالج (CPU) لتنفيذ مهامها (يكون أسرع من النوع الأول) ، ويطلق عليه أيضا (CPU Scheduler) .

3- المجدول متوسط الأمد (Medium-term scheduler):ومختصره M.T.S

يقوم بإزالة عملية معينة من الذاكرة الرئيسية بشكل مؤقت لفترة معينة وذلك حسب أولوية العملية ومن ثم إعادتها إلى الذاكرة واستئناف التنفيذ مرة أخرى وهذه يطلق عليها بالمبادلة (swapping).

<u>ملاحظة:</u>

1- يكون النظام متوازن عندما تكون عدد العمليات (processes) الداخلة للنظام تساوي عدد العمليات الخارجة من النظام .

2- المجدول طويل الأمد (Long-term scheduler) يسيطر على عدد العمليات الموجودة في النظام .

3- لتنظيم أداء النظام بأكمله، يحتاج مجدول الذاكرة إلى أن يحدد بحذر نوع العمليات وعدد العمليات التي سيحتفظ بها في الذاكرة، وتسمى بدرجة البرمجة المتعددة (Degree Of Multiprogramming) وهي عدد العمليات الموجودة داخل الذاكرة الرئيسية .

الجدولة الوقائية Preemptive Scheduling

نطلق عليها جدولة بتدخل أيضا بحيث تسمح بقطع تنفيذ العملية (Process) التي يتم تنفيذها حاليا في الوقت المناسب وان كانت في منتصف التنفيذ فتأخذ السيطرة منها وإعطائها للعملية الأخرى . برمجتها صعبة ولكنها أفضل من النوع الآخر.

الجدولة غير الوقائية Non-Preemptive Scheduling

نطلق عليها جدولة بدون تدخل تضمن للعملية أنها لن تترك المعالج حتى ينتهي وقت تنفيذها الحالي ثم تعيد المعالج للنظام .

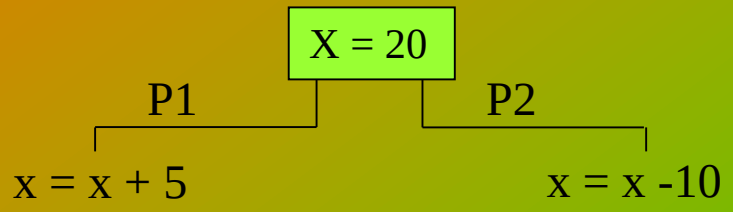
سؤال:

إن الجدولة الوقائية أفضل من الجدولة غير الوقائية ولكن فيها بعض المشاكل ، كيف؟

الجواب:

يمكن أن نأخذ المثال التالي ، لنفرض أن العملية (P1) لديها المعالج وتنفذ المهمة (x = x + 5) ولسبب ما فان نظام التشغيل سحب المعالج منها قبل خزن الناتج أعلاه ومنح المعالج إلى العملية الثانية (P2) لغرض تنفيذ المهمة (x = x - 10) مما يؤدي إلى ظهور المشاكل :

1- ضياع القيمة لـ (x) 2- حصول خلل بالتنفيذ



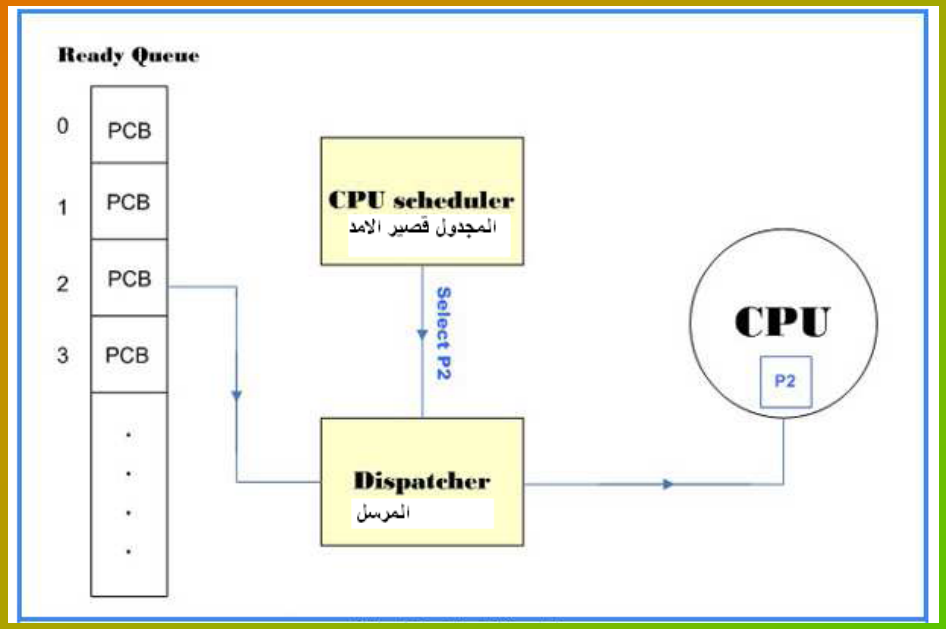
المرسل Dispatcher

عبارة عن وحدة (module) فوظيفتها تكمن في الجزء العملي أي أنها هي التي تقوم فعليا بالتنفيذ وتعطي سيطرة المعالج إلى العملية التي تم اختيارها من حالة الجاهز(ready) بواسطة المجدول قصير الأمد (S.T.S) وتحدث تحت نمط المراقب أو نمط النظام (monitor mode) , الشكل (32) يوضح عمل المرسل وعلاقته بالمجدول قصير الأمد ويتضمن عملها عدة أمور:

1- التبديل من عملية process إلى عملية أخرى ، وتحتاج إلى خزن الحالة القديمة للعملية وتحميل الحالة الجديدة وتحدث في نمط المراقب.

2- الانتقال إلى نمط المستخدم (user mode).

3- القفز إلى موقع أو عنوان مناسب في برنامج المستخدم لإعادة تنفيذ البرنامج.



الشكل (32) يبين العلاقة بين المجدول قصير الأمد والمرسل

وقت المرسل Dispatcher Latency

وهو الوقت الذي يقضيه ويستغرقه المرسل (dispatcher) لإيقاف تنفيذ عملية والبدء في تنفيذ عملية أخرى ، وهذا الوقت كلما كان أقل كلما كان النظام أفضل ، وان سرعة المرسل يجب أن تكون أقصى ما يمكن لأنه يستدعى عند كل اختيار لعملية من قبل المجدول قصير الأمد .

معايير الجدولة Scheduling Criteria

عند وجود عمليات كثيرة (Processes) في طابور الجاهز (ready queue) فإن المجدول قصير الأمد (S.T.S) سوف يختار إحداها للتنفيذ ، وهذه تعتمد على العديد من المعايير التي تحدد من هي العملية التي يجب تنفيذها ، ومن أهمها:

1- استغلال وحدة المعالجة المركزية (CPU utilization): (Maximum)

استغلال كل وقت وحدة المعالجة المركزية في تنفيذ العمليات ، أي أن تكون مشغولة بقدر الامكان ليتم استغلالها الاستغلال الامثل ، ومما يؤثر على هذا العامل هو عدد مرات التبديل التي تتم فكلما زاد هذا العدد كلما قل استغلال المعالج ، ما نريده ونصبوا إليه هو أعلى استغلال ممكن للمعالج إذ أننا لا نريد شغل وقت فراغ المعالج فقط بل شغله بما ينفع ولا تضييع وقته في عمليات التبديل .

2- الإنتاجية (Throughput) : (Maximum)

عدد العمليات المنفذة أو المنجزة خلال فترة زمنية محددة ، وكلما كان هذا العدد اكبر كلما كانت الخوارزمية أفضل .

3- الوقت الدوري (Turnaround time) : (Minimum) ومختصره (T.A.T)

الزمن المستغرق من قبل عملية معينة منذ لحظة دخولها إلى النظام إلى لحظة انتهاء تنفيذها وخروجها من النظام كلياً ، وكلما كان هذا الوقت أقل ما يمكن كلما كانت الخوارزمية أفضل . ويتضمن هذا الوقت الأوقات التالية :

- وقت الانتظار قبل الدخول إلى الذاكرة .
- الانتظار في طابور الجاهز (Ready queue) .
- وقت التنفيذ على وحدة المعالجة المركزية .
- تنفيذ عمليات الإدخال والإخراج .

ويمكن حساب الوقت من خلال المعادلة التالية :

الوقت الدوري = وقت انتهاء تنفيذ العملية – وقت وصول العملية .

4- زمن الانتظار (Waiting time) : (Minimum) ومختصره (W.T.)

الزمن الذي تستغرقها العملية داخل طابور الجاهز (Ready queue) بانتظار دورها في التنفيذ قبل دخولها إلى وحدة المعالجة المركزية (CPU) . ويمكن حسابه من خلال :

زمن الانتظار (W.T) = الوقت الدوري (T.A.T) - وقت التنفيذ (Execution time) للعملية.

5- زمن الاستجابة (Response time) : (Minimum)

يعتبر هذا الوقت من لحظة تنفيذ العملية حتى ظهور أول نتيجة لها بسرعة ، يستخدم هذا العامل عادة في الأنظمة التفاعلية (Interactive system) التي يكون بها المستخدم طرفاً .

نستنتج مما سبق أن نظام التشغيل يسعى إلى تحسين أداء المعالج بتحقيق الآتي :

- زيادة استغلال للمعالج .
- زيادة الإنتاجية .
- تقليل الوقت الدوري .
- تقليل زمن الانتظار .
- تقليل زمن الاستجابة .

أسئلة اختبارية :

س1: ما الغرض من جدولة العمليات الموجودة قي الذاكرة ؟

س2: ما هي اصناف العمليات بالاعتماد الى العمل او الوظيفة ؟ معززا بالرسم .

س3: عدد انواع المجدول Scheduler ؟

س4: الجدولة الوقائية افضل من الجدولة غير الوقائية ولكن فيها بعض المشاكل . كيف ؟ مع مثال.

س5: الوقت الدوري T.A.T يتضمن مجموعة اوقات . اذكر هذه الاوقات ؟