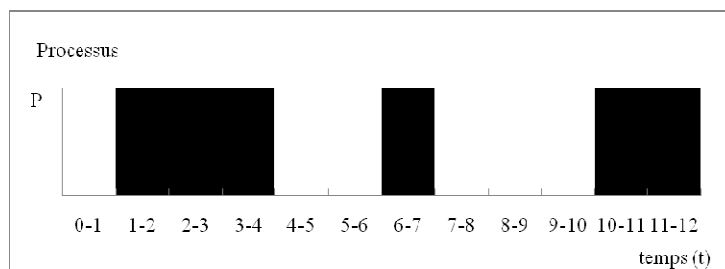


# ORDONNANCEMENT DANS LES SYSTEMES CLASSIQUES

Ce TD se compose d'une annexe A. Il est recommandé d'utiliser un tableur pour réaliser les différents calculs. On rappelle que les supports de cours sont disponibles à <http://mathieu.delalandre.free.fr/teachings/realtime/>

## 1. Paramètre des processus et exécution

Soit le diagramme d'occupation du processeur suivant, décrivant l'exécution du processus P. Les zones noires correspondent aux cycles actifs (i.e. où P est exécuté par le CPU).



Indiquer les valeurs suivantes: date de réveil (on considère ici  $r_0 = \text{start}$ ), capacité, temps d'attente et de réponse. Donner également sur l'intervalle de temps considéré (i.e. pour tout  $t$ ) la valeur de la capacité résiduelle  $C(t)$ .

## 2. Algorithmes non préemptifs à priorités statiques

Soit les tâches suivantes :

	<b>r0</b>	<b>C</b>	<b>Pr</b>
A	0	60	3
B	0	100	2
C	0	40	4
D	0	20	1
E	0	80	5

Donner les diagrammes d'occupation du processeur pour les algorithmes d'ordonnancement FCFS, SJF « non préemptif » et Priority Scheduling « non préemptif ». Dans le cas du FCFS, on considérera les id processus (A à E) comme critère d'ordonnancement. Indiquer les temps d'attente de chacune des tâches et le temps d'attente moyen pour chacun des algorithmes.

## 3. Algorithmes préemptifs à priorités statiques

Soit les tâches suivantes :

	<b>r0</b>	<b>C</b>	<b>Pr</b>
A	0	4	4
B	3	6	3
C	4	3	2

D	8	5	5
E	9	2	1

Donner les diagrammes d'occupation du processeur pour l'algorithme d'ordonnancement SJF « préemptif » et Priority Scheduling « préemptif ». Indiquer les temps d'attente de chacune des tâches et le temps d'attente moyen pour chacun des algorithmes.

#### 4. Algorithmes préemptifs à priorités dynamiques

4.1 Soit les tâches suivantes :

	r0	Pr
A	0	3
B	3	5
C	4	4
D	8	2

Donner le diagramme d'occupation du processeur pour l'algorithme d'ordonnancement Dynamic Priority Sheduling pour t [0-20]. Indiquer les temps d'attente de chacune des tâches et le temps d'attente moyen sur l'intervalle considéré.

4.2 Soit les tâches suivantes :

	r0	C
A	0	200
B	0	170
C	0	80
D	0	50

Donner le diagramme d'occupation du processeur pour l'algorithme d'ordonnancement Round Robin (commencer le tourniquet à la seconde tâche B) avec un quantum  $q = 40$ . Indiquer les temps d'attente de chacune des tâches et le temps d'attente moyen.

On se propose de trouver le quantum  $q$  optimum permettant un temps d'attente minimum pour cette configuration de tâches, avec  $q = [10-100]$  ( $q$  variant par palier de 10). Réaliser cette étude à l'aide du logiciel Cheddar (Annexe A).

#### A. Prise en main du logiciel Cheddar

Le logiciel peut être téléchargé à <http://beru.univ-brest.fr/~singhoff/cheddar/>

Mise en route

1. Définir un processeur *edit/update processeur*
2. Définir un espace d'adresse *edit/update adress space*, associer cette espace d'adresse au processeur
3. Configurer les tâches *edit/update task*, définir des tâches périodiques
4. Simuler l'ordonnancement (barre d'outils, dernière icône)
5. Sauvegarde du projet en cours *file/save xml project as* « nom.xml »

Voir <http://beru.univ-brest.fr/~singhoff/cheddar/ug/cheddar-r2.html#Ref1> pour plus de détails