

**Questions de cours: (7pts)**

- 1/- Expliquez le mécanisme de commutation de contexte.
- 2/- quels sont les types d'interruptions, distinguez les types synchrone de celle asynchrone.
- 3/- Les interruptions viennent remplacer la technique d'attente active. Quel sont les inconvénients de l'attente active. Justifier l'utilisation des interruptions.
- 4/- Le PCB c'est quoi, proposer une structure de donnée pour ce dernier. Est il swapé. Expliquer son utilisation lors de l'occurrence de l'interruption horloge dans les cas suivants :
  1. Monoprogrammation (un seul programme en mémoire).
  2. Multiprogrammation (avec réquisition du processeur).
- 5/- Donner le diagramme de transition d'état d'un processus dans les cas suivants :
  1. Monoprogrammation (un seul programme en mémoire).
  2. Multiprogrammation (avec réquisition du processeur).

**Exercice 1: Gestion de mémoire (7pts)**

Considérez un système de mémoire virtuelle ayant les caractéristiques suivantes : Taille d'une page = 1 KO. Taille de la mémoire centrale = 32 MO. Taille de la mémoire virtuelle = 512 MO. Utilisation combinée des techniques de pagination et de segmentation : l'espace d'adressage virtuel d'un processus est composé de segments contigus. Chaque segment peut contenir entre 1 et 128 pages. La numérotation des pages d'un segment est relative au segment. L'algorithme de remplacement de pages utilisé est LRU.

1/- Calculez le format d'une adresse virtuelle et le format d'une adresse physique en spécifiant le nombre de bits réservés pour chaque champ.

2/- Supposez un processus de 9 KO de segment de code et 3 KO de segment de données. Dans l'espace virtuel du processus, le segment de code est suivi du segment de données. Par conséquent, le segment de code débute à l'adresse 0 alors que celui des données débute à l'adresse 9216 relativement au début de l'espace d'adressage virtuel.

- Calculez l'adresse qu'occupe en mémoire principale une donnée se trouvant à l'adresse 10728, relative au début de l'espace d'adressage. Le segment de données du processus est chargé au complet en mémoire physique dans les cases contigus 4096, 4097 et 4098.

3/- Considérez la séquence de références de pages de code  $R=\{0, 1, 0, 1, 2, 3, 4, 2, 3, 4, 5, 6, 7, 8\}$  faite par le processus décrit en (2). Les opérandes référés par les instructions dans les pages 0, 1 et 2 se trouvent dans la page 0 du segment de données ; les opérandes des instructions des pages 3, 4 et 5 sont dans la page 1;

les opérandes des instructions des pages 6, 7 et 8 sont dans la page 2. Supposez que toutes les instructions du processus ont un opérande qui réfère en mémoire. Au départ, 4 case contigus sont alloués pour le code du processus à l'adresse X et 2 case contigus pour les données du processus à l'adresse Y. Il est à noter que les adresses X et Y ne sont pas nécessairement contiguës, et le chargement des pages dans les cases allouées est réalisé à la demande (aucun chargement préalable). De plus, aucune case supplémentaire n'est alloué au processus durant son exécution.

a/- Représentez l'état d'occupation de la mémoire centrale à chaque instant  $t_i$  où une page est chargée.

b/- Calculez le nombre de défauts de page générées par l'algorithme de remplacement de pages LRU. Ce nombre est-il optimal ?

### Exercice 2: Gestion des interruptions (7pts)

Dans un système mono programmé, on souhaite donner la possibilité de suivre l'évolution des événements, particuliers et à volonté, engendrés par l'exécution d'un programme utilisateur. Cette volonté s'exprime par deux services du superviseur, notamment :

**ACTTRACER (P, Q) :** Activer le tracé des événements. IL active périodiquement la procédure « P » de l'utilisateur. Cette périodicité « Q » est exprimée en période de l'horloge physique.

**DESACTRACE :** Stopper le tracé des événements. Il désactive l'exécution périodique de « P ». On note la procédure « P » est désactivée de manière automatique (si elle ne l'est pas) à la fin de l'exécution du programme. Schématiquement le déroulement du programme utilisateur est comme suite :

```

Begin           //
                SVC ACTTRACER (Proc1, 200)
                //
                SVC DESACTRACE
                //
                SVC ACTTRACER (Proc2, 100)
End.            //

```

- Le système n'est pas réentrant. Expliquez la notion de réentrance d'un système.
- On ne peut pas utiliser la commande *CALL*, pourquoi. Il faut utiliser uniquement la commande *LPSW* pour résoudre le problème d'appel de la procédure P. Expliquez le fonctionnement de l'instruction *LPSW*, quel est son type.
- Définir les règles (s'ils y en à) qu'on doit respecter lors de l'écriture des procédures utilisateur P.
- Ecrire l'ensemble de routine permettant de gérer ce système.

*Bonne chance.*