**Réponse aux questions parties 2 :**

1. **Comment vous êtes-vous réparti le travail ?**

* Sacha Conti -> mise en place du gestionnaire d'interruptions
* Thierry Mourao -> gestion du clavier
* Sylvain Thullen -> gestion du timer

**2. Votre kernel comporte-t-il des bugs ? Si oui, lesquels et comment pourriez-vous les**

**corriger ?**

* Aucun bugs n’ont été remarqué dans notre kernel

**3. Dans quel ordre vous avez initialisé les différents points ci-dessus dans votre kernel ?**

**Justifiez.**

* 1. *VGA* -> On init d’abord le mode video avant de passer en mode protégé parce qu’on va devoir afficher les informations
  2. *GDT* -> Init de la GDT. On a besoin d’allouer la mémoire pour l’utiliser après.
  3. *PIC* -> On remappe les interruptions avant d’initialiser l’IDT
  4. *IDT* -> Initialisation de l’IDT avant le timer et le clavier car les 2 appelle une IRQ
  5. *TIMER*
  6. *KEYBOARD* -> L’ordre de l’init du clavier ou du timer n’a pas d’importance ici.

**4. Pourquoi remappe-t-on les IRQ 0 à 7 aux interruptions 32 à 39 ?**

* Car les 32 premières sont les interruptions du processeur Intel

**5. Que se passerait-il si on ne le faisait pas ?**

* On rentrerait en conflit entre nos interruptions et celle du processeur, ainsi on ne saurait si le processeur a un problème.

**6. Comment pouvez-vous tester que votre gestionnaire d'interruption pour les exceptions**

**fonctionne correctement ?**

* On peut volontairement faire une division par 0 et générer une exception.
* Ou faire une *segmentation fault* avec un pointeur

**7. Quelles exceptions avez-vous pu générer et comment avez-vous fait ?**

* Une exception de division par 0 en en faisant une volontairement dans les tests.

**8. Quelle taille de buffer clavier avez-vous choisie et pourquoi ?**

* 32 car on a un système 32bits et que l’on a pas besoin d’un buffer immense pour le clavier

**9. Comment pouvez-vous causer une situation de buffer plein quelle que soit la taille du**

**buffer (dans les limites du raisonable) ?**

* Il suffit de laisser appuyer sur une touche