

Test de Primalité

Jean-Didier Pailleux - Robin Feron - Romain Robert - Damien Thenot -
Maxence Joulin

UVSQ

11/01/2018

- **Nombre Premier** : Entier divisible par 1 et lui-même.
pause
- **Test Probabiliste** : Test avec marge d'erreur très faible mais rapide.
- **Test Deterministe** : Test fiable mais plus lent.
- 2^{64} : Taille maximale des nombre a tester \Rightarrow Unsigned long long int
- **Nombre Hautement Composé** : Entier qui possède strictement plus de diviseur que les nombres qui le précède

- 1 Etat de l'art
- 2 Implémentation
- 3 Analyse des Résultats
- 4 Bilan Technique
- 5 Conclusion

- 1 Etat de l'art
- 2 Implémentation
- 3 Analyse des Résultats
- 4 Bilan Technique
- 5 Conclusion

Qui ?

- 1 Etat de l'art
- 2 Implémentation**
- 3 Analyse des Résultats
- 4 Bilan Technique
- 5 Conclusion

Langages de programmation :

- Le langage C++ : langage adapté pour la programmation procédurale et pour la programmation orientée objet + bibliothèques pour les grands entiers supérieur , chronométré et autre.
- Le langage Shell

Langages de programmation :

- Le langage C++ : langage adapté pour la programmation procédurale et pour la programmation orientée objet + bibliothèques pour les grands entiers supérieur , chronométrer et autre.
- Le langage Shell

Outils

- La bibliothèque GMP : manipulation de nombres supérieur à 2^{64} .
- La bibliothèque NTL : fonctions pour l'arithmétique modulaire disponible.
- CMkake : pour la compilation du projet.
- Github : dépôt du projet + travail collaboratif.

- 1 Etat de l'art
- 2 Implémentation
- 3 Analyse des Résultats**
- 4 Bilan Technique
- 5 Conclusion

Lancement d'une phase de test :

- Appel d'un script bash ou lancement en ligne de commande.
- Script bash => test sur un fichier ou plage de valeurs ou test normal.
- **Options :**

a : Tous les algorithmes

e : Euclide (computation bound)

m : Crible d'eratosthene

i : Miller-Rabin

H : Nombre hautement composé def

k : AKS

o : Modulo (computation bound)

p : Pocklington

h : Nombre hautement composé naive

- Lequel utiliser ?
- Itération ?
- Combien de nombre ?
- Donner les nombres

Analyse des Résultats

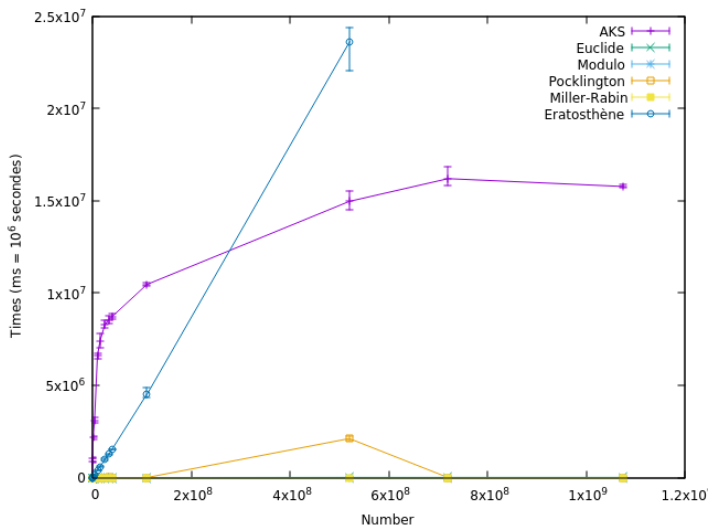


FIGURE – Évolution du temps d'exécution pour les 6 tests de primalité (AKS, Pocklington, Miller-Rabin, Euclide, Ératosthène et Modulo).

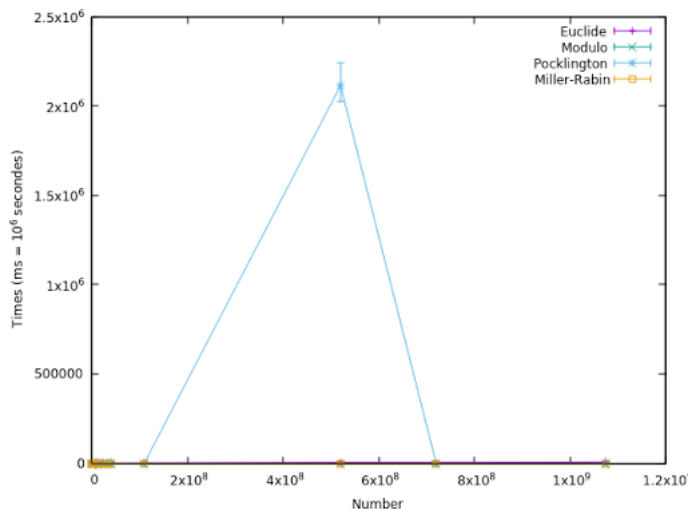


FIGURE – Zoom de la figure 1 sur (Pocklington, Miller-Rabin, Euclide et Modulo).

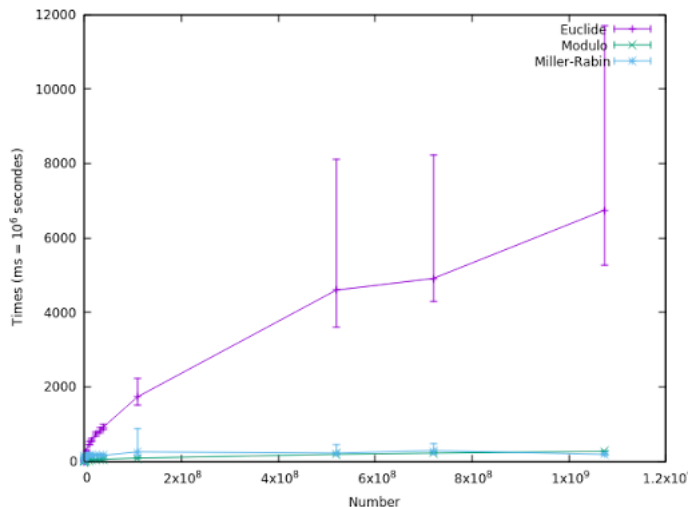


FIGURE – Zoom de la figure 1 sur (Miller-Rabin, Euclide et Modulo).

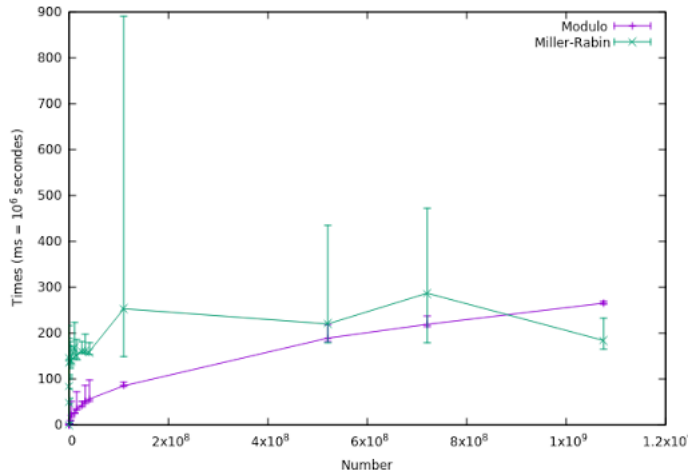
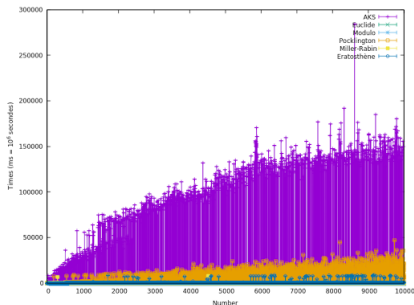
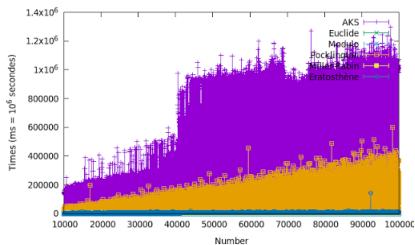


FIGURE – Zoom de la figure 1 sur (Miller-Rabin et Modulo).

Figures - Évolution du temps d'exécution pour les 6 tests de primalité sur une plage de données



Plage [1 : 10000]
100% de réussite pour Miller-Rabin



Plage [10000 : 100000]
100% de réussite pour Miller-Rabin

- 1 Etat de l'art
- 2 Implémentation
- 3 Analyse des Résultats
- 4 Bilan Technique**
- 5 Conclusion

- **Eratosthène** : Création d'un tableau de taille $N+1$ dans le crible \Rightarrow limité au niveau de la RAM pour N grand sur nos machines. Complexité de N pour le remplissage de la liste `memory_bound`.
- **Euclide** : Effectue $\sqrt{2^{\log_2(n)}}$ divisions euclidiennes \Rightarrow Exécution en temps exponentiel
- **Pocklington** : Limite causé par la factorisation du nombre $N-1 \Rightarrow$ factorisation très longues pour N très grand.
- **Miller-Rabin** : Résultats faux dans certains cas + Nombre d'itérations demandé élevé pour un meilleur résultat \Rightarrow augmentation du temps d'exécution.
- **AKS** : Avantage : Sa complexité en $\log(n)^{12}$.
Inconvénient : Utilisation de NTL qui effectue des vérifications superflue + Implémentation compliquée.

Plan

- 1 Etat de l'art
- 2 Implémentation
- 3 Analyse des Résultats
- 4 Bilan Technique
- 5 Conclusion**

- De grands temps de calculs pour les tests déterministes.
- Méthodes naïves efficaces pour les petits nombres.
- Tests probabiliste une bonne idée ?
- **Possibilités de parallélisation** : Une ouverture sur des techniques de calcul en parallèle pourrait être appliqués.

Tableau de répartition du travail :

Tâches	Jean-Didier	Maxence	Romain	Robin	Damien
Eratosthène/Memory Bound	x				
Euclide/Computation Bound		x			
AKS			x		
Pocklington					x
Miller-Rabin				x	
Highly Composite	x				
Cmake	x	x			
Script/main	x	x			