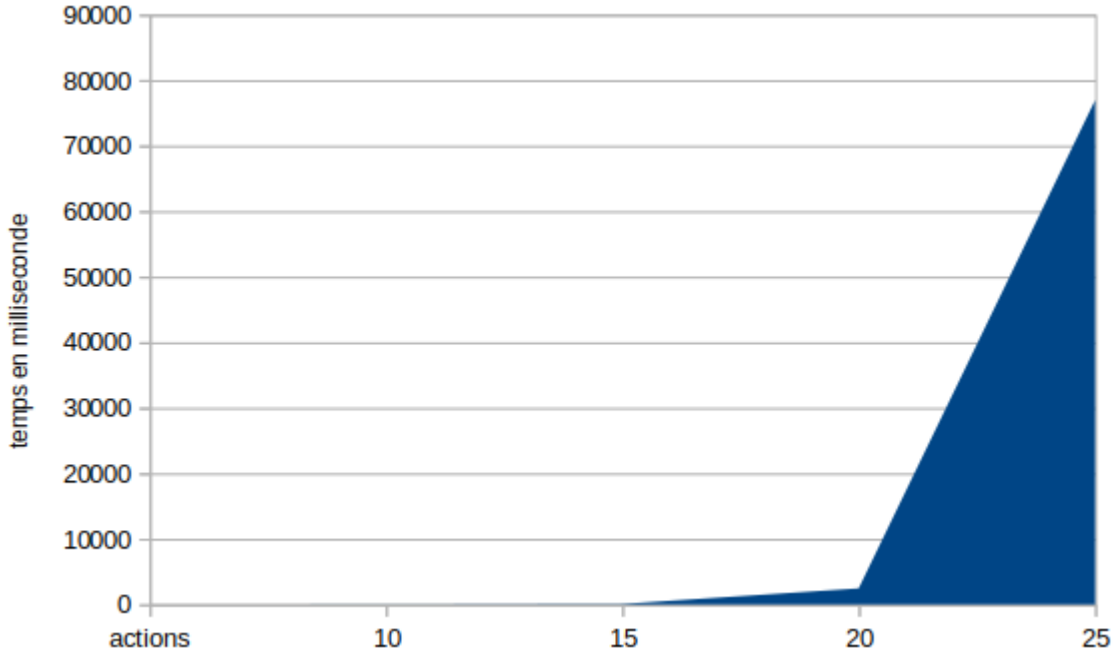
# Force Brut

Avec la méthode de force brute notre algorithme vas tester l'intégralité des possibiltées.

Donc pour une liste d'actions comprenant 5 actions nous aurons 32 possibiltées soit **p = 2^5**. si nous ajoutons une actions de plus nous doublons les possibilitées **p = 2^6** soit 64.

Notre Résultat est exponentiel et avec une combinaison de 20 actions nous aurons donc 1.048.576 possibilitées.

Après éxécution de l'algorithme écrit avec un dataset de 20 actions l'éxécution de celle ci prend 2.34 secondes si nous ajoutons 2 actions en plus l'éxecution du script prend maintenant 9.16 secondes



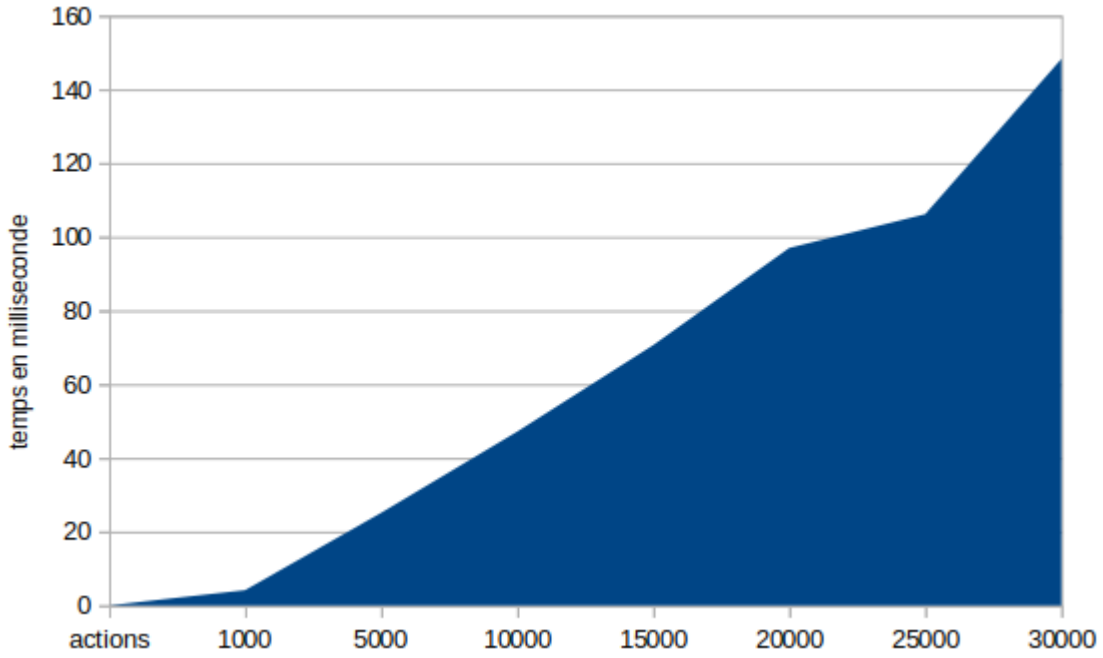

# Algorithme Optimisé

Ici nous n'allons plus prendre tous les possibilitées en charge, mais nous allons trié le data set en prenant en charge le meilleur ratio (bénéfice / cout de l'action) pour chaque actions.

Après avoir bouclé sur le data set en vérifiant que l'ajout de l'action correspond bien a notre budget (et si ce n'est pas le cas la boucle seras cassé) l'action seras ajouté dans le portefeuilles.

Notre algorithme (glouton) n'est plus exponentielle mais linéaire.

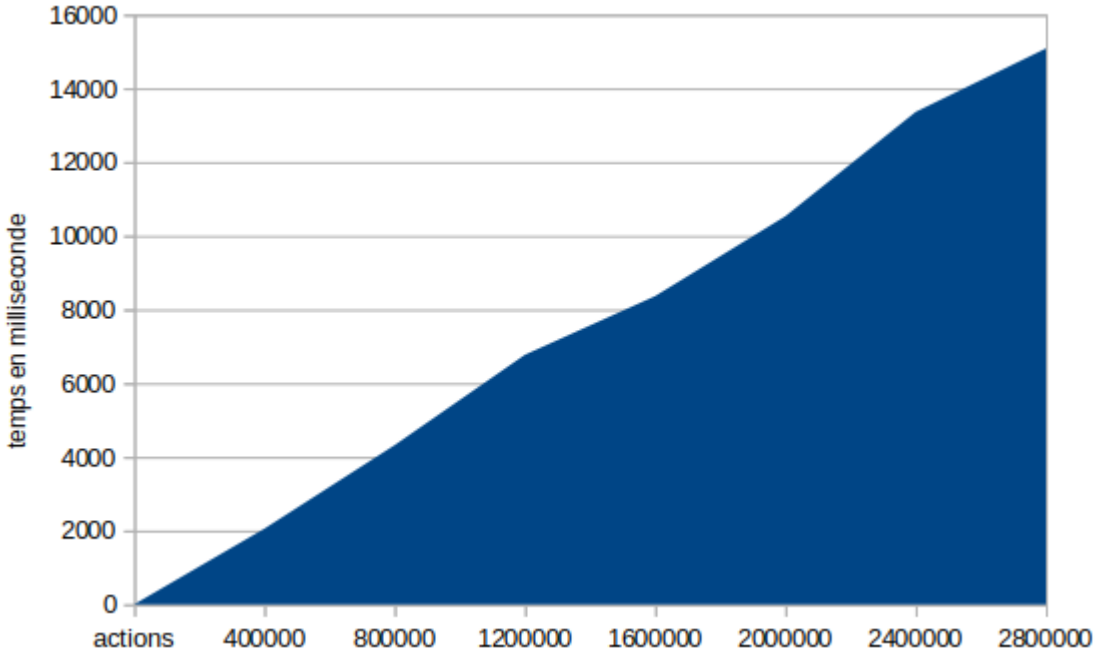
```
list = ranger(list_action par bénéfice/cout de l'action)
pour chaque action dans list:
        si prix > 0 et ((cout_portefeuille + prix) <= 500):
                ajout_portefeuille(action)
        si cout_portefeuille >= 500:
            on stop la boucle
```




# Limites de l’algorithme

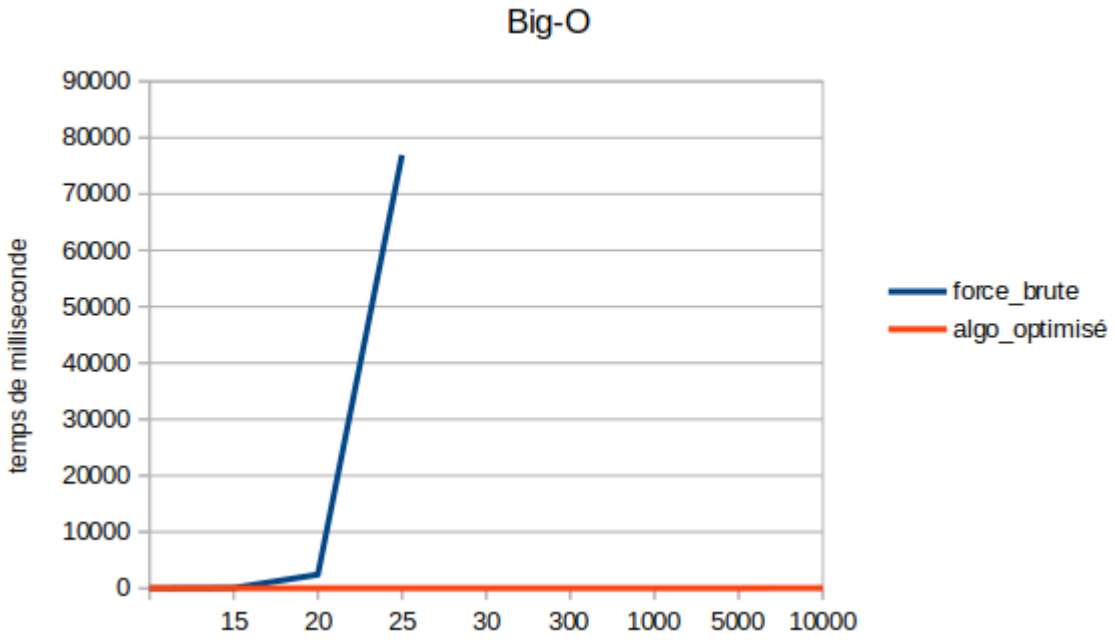
Pour pouvoir traiter un gros portefeuille d’action (plus de 100.000 actions) l’algorithme choisi est un algo glouton.

Les prix des actions ou les bénéfices de ceux-ci ne joue pas sur le temps d’exécution.

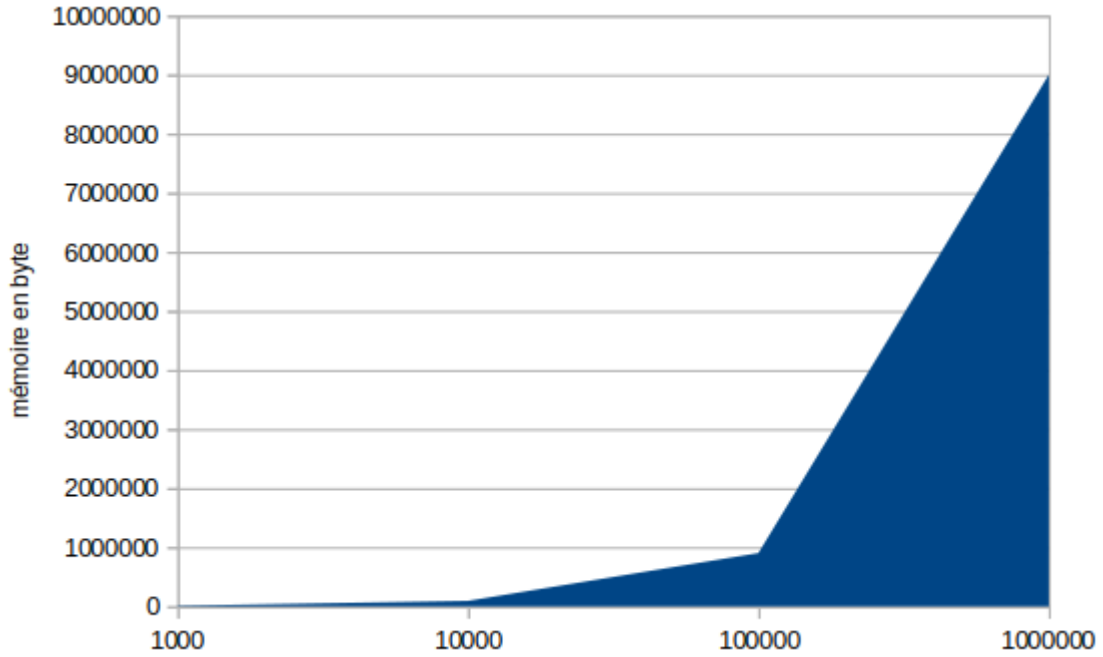



Pour garder l’exécution a moin de 1 secondes il faudrait avoir moins de 200.000 actions. Sur ce graphique on peut voir qu’avec 2.000.000 d’actions il faut 15 secondes sur l’exécution ceux qui peut être considéré comme raisonnable

# Notation Big-O,




la complexité temporelle???




Pour l'utilisation de la mémoire on peut voir sur le graphique qu'avec 1 millions d'action l'algorithme utilise 8.58 Mo

## dataset1_Python+P7.csv

| vianney | sienna |
|---|---|
| Share-XJMO, price: 9.39€ | Share-GRUT, price: 498.76€ |
| Share-MTLR, price: 16.49€ | |
| Share-KMTG, price: 23.21€ | |
| Share-LRBZ, price: 32.9€ | |
| Share-GTQK, price: 15.4€ | |
| Share-WPLI, price: 34.64€ | |
| Share-GIAJ, price: 10.75€ | |

| vianney | sienna |
|---|---|
| Share-GHIZ, price: 28.0€ | |
| Share-IFCP, price: 29.23€ | |
| Share-ZSDE, price: 15.11€ | |
| Share-FKJW, price: 21.08€ | |
| Share-NHWA, price: 29.18€ | |
| Share-LPDM, price: 39.35€ | |
| Share-QQTU, price: 33.19€ | |
| Share-USSR, price: 25.62€ | |
| Share-EMOV, price: 8.89€ | |
| Share-LGWG, price: 31.41€ | |
| Share-SKKC, price: 24.87€ | |
| Share-QLMK, price: 17.38€ | |
| Share-UEZB, price: 24.87€ | |
| Share-CBNY, price: 1.22€ | |
| Share-CGJM, price: 17.21€ | |
| Share-EVUW, price: 4.44€ | |
| Share-FHZN, price: 6.1€ | |
| Share-MLGM, price: 0.01€ | |
| total: 499.94€ | total: 498.76€ |
| profit: 198.51€ | profit: 196.61€ |

## dataset2_Python+P7.csv

| vianney | sienna |
|---|---|
| Share-PATS, price: 27.7€ | Share-PATS 27.70€ |
| Share-JWGF, price: 48.69€ | Share-JWGF 48.69€ |
| Share-ALIY, price: 29.08€ | Share-ALIY 29.08€ |
| Share-NDKR, price: 33.06€ | Share-NDKR 33.06€ |
| Share-PLLK, price: 19.94€ | Share-PLLK 19.94€ |
| Share-FWBE, price: 18.31€ | Share-FWBE 18.30€ |
| Share-LFXB, price: 14.83€ | Share-LFXB 14.83€ |
| Share-ZOFA, price: 25.32€ | Share-ZOFA 25.32€ |
| Share-ANFX, price: 38.55€ | Share-ANFX 38.54€ |
| Share-FAPS, price: 32.57€ | Share-FAPS 32.57€ |
| Share-LXZU, price: 4.24€ | |
| Share-XQII, price: 13.42€ | Share-XQII 13.42€ |
| Share-ECAQ, price: 31.66€ | Share-ECAQ 31.66€ |
| Share-JGTW, price: 35.29€ | Share-JGTW 35.29€ |
| Share-IXCI, price: 26.32€ | Share-IXCI 26.32€ |
| Share-DWSK, price: 29.49€ | Share-DWSK 29.49€ |
| Share-ROOM, price: 15.06€ | Share-ROOM 15.06€ |
| Share-VCXT, price: 29.19€ | |
| Share-YFVZ, price: 22.55€ | Share-YFVZ 22.55€ |

| vianney | sienna |
| --- | --- |
| Share-OCKK, price: 3.16€ | |
| Share-JMLZ, price: 1.27€ | |
| Share-DYVD, price: 0.28€ | |
| | Share-VCAX 27.42€ |
| total = 499.98€ | total = 489.24€ |
| profit = 197.77€ | profit = 193.78€ |