

pushswap correction

Vérifions differents cas de figure pour controler que ce push\_swap marche bien.

La gestion des erreurs

Effectuez des tests pour tenter de mettre en defaut la gestion d'erreurs : - "push\_swap" avec des paramètres non numériques - "push\_swap" avec 2 fois le même paramètre - "push\_swap" avec des paramètres plus grand que l'int Tout marche comme il faut ? Les messages d'erreurs sont formates conformement au sujet ?

ve so

Test sans rien

Testez le pushswap sans aucun parametre. Le programme doit rendre la main. On tolere le fait qu'il y ai une ligne vide, correspondant a aucune instruction a effectuer.

ve so

Test avec un seul element

Testez avec un unique parametre valide (la gestion des erreurs c'etait dans la section precedente). On doitavoir 0 operation a effectuer.

ve so

Est-ce que cela marche ?

Lancez le tri avec un petit nombre de parametres positifs (genre 4 ou 5). Verifiez que le resultat propose permet bien de trier les elements sur la pile a, du plus petit au sommet de la pile au plus grand.

ve so

Un peu de negatif

Effectuez plusieurs tests, toujours avec un petit nombre d'elements, mais avec certains / tous qui sont negatifs. Ca marche toujours ?

ve so

Gros volume

Testez enormement de parametres. Est-ce que le programme fournit un resultat qui semble logique par rapport au nombre d'elements ? Dans un delairaisonnable ? Vous pouvez utiliser par exemple :

./push\_swap ruby -e "puts (-1000..1000).to\_ashufflejoin('')"

**Ou encore**

./push swap ruby -e "puts (-5000.5000).to\_a reverse.insert(rand(8000) + 1000, 10001).join('')"

ve so

Minimum d'instructions

Pour rappel, le sujet vous demande de fournir une solution contenant un minimum d'instruction. Regardez quel type de structure de données est utilisé pour le stockage des entrées avant affichage.

Y a rien a faire

Testez une suite d'elements deja triee en parametre. Si vous n'avez pas comme resultat0 instruction a effectuer, c'est pas bon.

ve so