

Caso 00: casoDelEnunciado

Verifica que calcula de manera correcta la cantidad de apps a sacar en el caso de un flujo normal.

casoDelEnunciado.in	casoDelEnunciado.out
10 100 42 2 50 10 1 50 30 24 18 23	3

Caso 01: consideraElMenorGrupoDeApps

Verifica que tome el grupo con la cantidad mínima de aplicaciones, aunque haya varios grupos que lleguen a la cantidad de mb necesarios.

consideraElMenorGrupoDeApps.in	consideraElMenorGrupoDeApps.out
12 60 1 1 50 1 1 50 1 1 20 20 20 1	3

Caso 02: unaSolaAplicacion

Busca detectar un error en caso de que tenga una sola aplicación y alcance la cantidad necesaria de Mb para instalar la aplicación A.

unaSolaAplicacion.in	unaSolaAplicacion.out
1 10 10	1

Caso 03: noChequeaPrimerApp

Busca detectar un error en caso de que el grupo de aplicaciones a desinstalar incluya a la primera pero esta no sea tomada en cuenta.

noChequeaPrimerApp.in	noChequeaPrimerApp.out
11 60 20 20 20 1 1 50 1 1 50 1 1	3

Caso 04: noChequeaUltimaApp

Busca detectar un error en caso de que el grupo de aplicaciones a desinstalar incluya a la ultima pero esta no sea tomada en cuenta.

noChequeaUltimaApp.in	noChequeaUltimaApp.out
11 60 1 1 50 1 1 50 1 1 20 20 20	3

Caso 05: fatiga

Verifica que en caso de que el programa reciba una entrada con 50000 aplicaciones instaladas y la cantidad de mb requeridos de la aplicación A sean 1000, funcione correctamente.

fatiga.in	fatiga.out
05.in	1000

