

## Taak 1.2 – Schapen winnen (sheep) (100 ptn)

Boer Azamat neemt deel aan een spelprogramma op de televisie, en naast het gebruikelijke vertier met playback zangers kan hij ook nieuwe schapen voor op zijn boerderij winnen.

Het spel gaat als volgt. Eén voor één worden de schapen die Azamat kan winnen binnengebracht in de studio. Als Azamat dan kan zeggen hoeveel paren schapen er in de foute volgorde staan – dit is: het schaap dat eerder is binnengekomen is groter dan het andere schaap – van alle schapen die er ondertussen in de studio staan, wint hij ook dat nieuwe schaap. Als hij een fout antwoord geeft, of niet op tijd antwoordt, stopt het spel.

Azamat kan niet heel goed tellen, maar omdat hij wel heel vertrouwd is met schapen kan hij wel exact de grootte van een schaap inschatten. Hij maakt met jou de volgende afspraak: hij geeft je de grootte van het nieuwe schaap, en jij geeft het juiste antwoord op het spel. Als Azamat alle schapen wint, krijg jij voor de komende 3 jaar een kwart van de opbrengst van hun wol.

### Taak

Help Azamat het spelprogramma te winnen.

### Limieten en beperkingen

- $1 \leq N \leq N_{MAX}$ , het totaal aantal schapen;
- $0 \leq M \leq M_{MAX}$ , de maximale grootte van een schaap, in een of andere lengtemaat;
- Alle schapen hebben een verschillende grootte.

	$N_{MAX}$	$M_{MAX}$
Subtaak A (12 ptn)	200	1 000
Subtaak B (18 ptn)	3 000	100 000
Subtaak C (50 ptn)	50 000	1 000 000
Subtaak D (20 ptn)	50 000	$10^{12}$

Maximale uitvoeringsduur: **3 seconden**. Geheugenlimiet: **512 MB**.

### Te implementeren functie

**C++:** `long long newSheep(long long N, long long M, long long i, long long size)`

**Java:** `long newSheep(long N, long M, long i, long size)`

Deze functie neemt als argumenten:

**N** Het totaal aantal schapen

**M** De maximale grootte van een schaap

**i** Het nummer van het schaap, dus in opeenvolgende oproepen 1, 2, 3, ...

**size** De grootte van het nieuwe schaap

Deze functie moet als resultaat het antwoord op de spelvraag teruggeven als een geheel getal.

### Voorbeeld

Hier volgt een mogelijk scenario:

$N$	$M$	$i$	$size$	Huidige schapen	resultaat	<i>foute</i> paren
4	10	1	8		0	
4	10	2	5	8	1	(8, 5)
4	10	3	7	8, 5	2	(8, 7), (8, 5)
4	10	4	2	8, 5, 7	5	(8, 7), (8, 2), (5, 2), (8, 5), (7, 2)

### Praktische opmerkingen

Deze taak is interactief. Je zal dus niet de automatische compilatie en test van gedit kunnen gebruiken voor deze taak. Volg de onderstaande stappen om je programma manueel te compileren en te testen. Vraag hulp aan een toezichter indien nodig.

- Open een terminal (de applicatie LXTerminal)
- Om de lijst met bestanden in een folder weer te geven, gebruik het commando `ls`
- Om van folder te veranderen, naar het skelet van deze taak, gebruik `cd folder_naam`

Voor degenen die C++ gebruiken, zijn de commando's de volgende:

- Compilatie: `g++ -std=c++11 -Wall -Wextra -Wshadow grader.cpp sheep.cpp`
- Uitvoering: `./a.out < input.txt`

Voor wie Java gebruikt, zijn de commando's als volgt:

- Compilatie: `javac grader.java`
- Uitvoering: `java grader < input.txt`

De uitkomsten van de opeenvolgende oproepen naar jouw functie zullen in de console op aparte regels afgeprint worden.

### Opmerkingen

- Je moet enkel het bestand `sheep.cpp` of `sheep.java` indienen. Je moet de functie `newSheep` implementeren, zoals hierboven beschreven.
- Print niets uit naar de console in het programma dat je indient.