

# beCP 2022

## Taak 2.1: Secret Agents (secretagents)

---

Author: Damien Leroy

Maximale uitvoeringsduur: 2 s Geheugenlimiet: 64 MB

---

*Opmerking: Dit is een interactieve taak. Kijk de speciale instructies om je programma te implementeren, compileren en testen na. Aarzel niet om een begeleider om hulp te vragen.*

Je geheime agenten zijn nu actief op in het oorlogsgebied. Om hun aanwezigheid bekend te maken, gebruiken ze een communicatiesysteem dat hun agent-id door uitzendt door radiogolven. Om tracking te vermijden, kan de zender fouten introduceren in het id dat het uitstuurt.

Jouw taak is om de code te schrijven van de codeermachine, die voor de naam van een agent een 6-cijferig id geeft, en van de decodeermachine, die een voor een (mogelijks aangepast) id de naam van de bijhorende agent geeft.

### Te implementeren functies

---

C++	<code>vector&lt;int&gt; encode(vector&lt;string&gt; agents)</code>
<b>return</b>	Gegeven een lijst van lengte $N$ met namen van agenten (allemaal verschillend), geeft een lijst met id's terug. Deze functie wordt slechts één keer opgeroepen. Een lijst van lengte $N$ van agent-id's (6-cijfige getallen). Het id op index $i$ correspondeert met de agent <code>agents[i]</code> .

---

C++	<code>string decode(vector&lt;string&gt; agents, int id)</code>
<b>return</b>	Gegeven een lijst met namen van agenten van lengte $N$ in dezelfde volgorde als in <code>encode</code> en een id, geeft de naam van de agent die correspondeert met de gegeven id (mogelijks met een fout in). De id komt altijd overeen met een bestaande agent (met een eventuele aanpassing). Deze functie wordt ten hoogste 60000 keer aangeropen, altijd na <code>encode</code> . De naam van de agent die overeen komt met de gegeven id.

---

### Algemene limieten

- $2 \leq N \leq 1000$

## Bijkomende beperkingen

Subtaak	Punten	Beperkingen
A	40	$N \leq 100$ , de <i>id</i> 's om te decoderen zijn <b>niet</b> aangepast
B	40	$N \leq 100$ , in de <i>id</i> 's om te decoderen is hoogstens 1 cijfer anders
C	20	in de <i>id</i> 's om te decoderen is hoogstens 1 cijfer anders

## Technical details

Zodat je lokaal kunt testen, voorzien we een grader (niet dezelfde als die op de server) die je vrijelijk kunt aanpassen. Deze grader moet **niet** ingediend worden. Je kunt je code samen met de grader lokaal compileren met:

```
g++ -std=c++11 -Wall -Werror -Wshadow grader.cpp secretagents.cpp
```

### Voorbeeld 1 (deeltaak A: id's zijn niet aangepast)

Stel dat `encode` wordt aangeroepen met deze `agents` lijst:

Rohaam Wilkerson	Anwar Tierney	Caroline Scott
------------------	---------------	----------------

Dan kun je, bijvoorbeeld, dit teruggeven:

1 (= 000001)	900990	249234
--------------	--------	--------

Vervolgens kan de `decode` functie verschillende keren aangeroepen worden, met de zelfde `agents` als hierboven en de volgende `id`'s:

id	verwacht antwoord
900990	Anwar Tierney
1	Rohaam Wilkerson

### Voorbeeld 2 2 (subtask B/C: ids may be altered)

Stel dat `encode` wordt aangeroepen met deze `agents` lijst:

Murphy Aguirre	Kierran Heath	Anton Russo	Claudia Rogers
----------------	---------------	-------------	----------------

Dan kun je, bijvoorbeeld, dit teruggeven:

123456	990000	30 (= 000030)	666666
--------	--------	---------------	--------

Vervolgens kan de `decode` functie verschillende keren aangeroepen worden, met de zelfde `agents` als hierboven en de volgende `id`'s:

id	verwacht antwoord
990400	Kierran Heath
800030	Anton Russo
0 (= 000000)	Anton Russo
666666	Claudia Rogers

**Hints: Cijfers uit een getal isoleren**

Om een specifiek cijfer uit een getal te halen, moet je de gehele deling (/) en de modulo (%) operators gebruiken. De gehele deling is een deling waarin het deel na de comma wordt weggelaten. Bijvoorbeeld  $1234/100$  geeft het geheel getal 12. De modulo operator geeft de rest van de deling terug, dus  $1234\%100$  geeft 34.

Door deze twee te combineren kan je een specifiek cijfer uit een getal halen, bijvoorbeeld  $(1234/100)\%10$  geeft 2,  $(1234/10)\%10$  geeft 3.

Op dezelfde manier kun je getallen uit cijfers opbouwen door vermenigvuldiging. Als een getal wilt maken met twee keer 12, dan moet je gewoon dit doen:  $12 * 100 + 12$ . Als je een 5 van voor wilt toevoegen, tel dan er  $5 * 10000$  bij op, enz.