

Vingla hem

När festprissen Bengan ska gå hem vinglar han omkring på vägen och ramlar till slut ner i diket. Simulera hans avbrutna hemfärd med ett program. Antag att han slumpmässigt vinglar åt vänster eller höger för varje steg. Programmet ska avslutas när han hamnat i diket.

Arrayer får inte användas. Vägens bredd bestämmer du själv.

Programkörningsexempel:

Bengan är ute och går.

```
|      *      |
|        *     |
|       *      |
|      *       |
|     *        |
|    *         |
|   *          |
|  *           |
| *            |
```

Och nu ramlade han ned i diket!

Tips: Använd ett heltal för att hålla reda på var på vägen Bengan befinner sig. Om talet blir för stort eller litet har Bengan drullat ned i diket.

Slumpen

Detta med slumpen kan fungera olika med olika kompilatorer och plattformar.

`srand(time(NULL));` //initierar ett startvärde för "slumpen".

Slumpserien blir alltid lika med samma startvärde. Så vi använder tiden. Den ändras ju hela tiden.

Funktionen `rand()` returnerar ett slumptal mellan 0 och `RAND_MAX`. `RAND_MAX` är beroende av det system man kör på.

EX:

```
srand(time(NULL));
```

Om NULL

funktionen `time()` vill ha en pekare som argument. NULL är en pekare som inte pekar någonstans. Men i alla fall en pekare.

Och en pekare är adressen i minnet till platsen där något finns.

EX:

```
int i; //Kan innehålla en int
int *p; //Kan innehålla en pekare (adressen) till en int;
cout << (rand() % 10); //Levererar slumpvärden mellan 0 och 9
```

Sök information om:

`srand()`, `rand()` och `time()`.

Funktionen `"setw(int)"` kan vara bra att använda. Den tar alltså en `int` som argument.