Specifikation: Hitta öppning i en mur.
Förutsättningar: Vi vet inte i vilken riktning muren ligger. Vi vet inte avståndet till muren. Muren är oändligt lång i båda riktningarna.

Lösning:
Först: hitta muren.
Sedan: hitta öppningen i muren
Gå i en spiralrörelse till dess Robbie når fram till muren.
Det kan t.ex. se ut så här. Men det finns också andra sätt för Robbie att gå i en spiral, man kan öka antal steg vid varje sväng på andra sätt.
I figuren är olika faser markerade, 1-3.
(1) Robbie kollar om muren finns alldeles framför
(2) om ingen mur tar Robbie ett (bestämt) antal steg framåt
(3) sedan vrider sig Robbie 90° åt höger

(1) - (3) upprepas tills muren hittas. Eftersom vi inte har någon aning om avståndet kan vi sätta ”upprepa oändligt många gånger”, och hoppa ur loopen när målet nåtts.
(1) Robbie kollar om muren finns alldeles framför
(2) Om ingen mur tar Robbie ett (bestämt) antal steg framåt
(3) Sedan vrider sig Robbie 90° åt höger

(1) - (3) upprepas tills muren hittas.
Eftersom vi inte har någon aning om avståndet kan vi sätta upprepa oändligt många gånger, och hoppa ur loopen när målet nåtts.

Procedur Hittamuren
{ n:=0
  Upprepa ∞ ggr
  { Om HinderF?
    Exit Hittamuren
    n:=n+1
  }
  Upprepa n ggr Steg
  Om HinderF?=falskt
    Vrid (+1)
    Upprepa n ggr Steg
    Om HinderF?=falskt
    Vrid (+1)
}
(1) Robbie kollar om muren finns alldeles framför
(2) om ingen mur tar Robbie ett (bestämt) antal steg framåt
(3) sedan vrider sig Robbie 90° åt höger

(1) - (3) upprepas tills muren hittas.
Eftersom vi inte har någon aning om avståndet kan vi sätta upprepa oändligt många gånger, och hoppa ur loopen när målet nåtts.

Antalet steg som tas i exemplet blir
1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 8 (+ 2 "slirsteg),
nästan summan av de 10 första heltalen,
vilket också kan sägas vara problemets komplexitet.
Kan vi uttrycka detta mer generellt?
Vi tar hjälp av s.k. aritmetiska summor, se förklaring nästa bild.
Aritmetiska summor

Vad blir
1 + 2 + 3 + 4 + 5 + 6 + . . . + 100?
Vi lägger ihop den dubbla summan, och skriver på följande sätt (grupperar början och slutet av talserierna):
(1 + 100) + (2 + 99) + (3 + 98) + . . . + (99 + 2) + (100 + 1)
Vi ser att varje summa inom parentes blir 101. Vi har 100 sådana summor.

Det innebär att 100 * (100+1)/2 är ett sätt att beskriva summan av alla tal från 1 till 100.

Detta kan generaliseras till n * (n + 1)/2 = (n^2 + n)/2
Testa gärna formeln på exempelvis n=3, n=6.
N= antal steg
I exemplet är N= 10 när muren hittas, avståndet från Robbie till muren är ungefär N/2 eller D

Avståndet till muren: D
N ≈ 2D
Aritmetisk summa:
n * (n+1)/2
Vi byter ut n mot 2D i formeln:
2D * (2D+1)/2=D * (2D+1)= 2D^2+D
≈2D^2
Nu är Robbie framme vid muren, men hur ska han hitta till öppningen i muren?
Algoritm:
#steg nollställs
vrid Robbie åt vänster (ex.vis)
Upprepa ∞ ggr
1) Gå till vänster
2) Vrid 180°
3) Gå till höger

Gå åt vänster
1) Räkna upp #steg
2) Upprepa #steg steg
3) Kolla öppning höger om öppning-KLAR

Gå åt höger
1) Räkna upp #steg
2) Upprepa #steg steg
3) Kolla öppning vänster om öppning-KLAR
När Robbie går fram och tillbaka, till höger och vänster, ökas antal steg med 1 i varje vända. Det ger oss alltså ett läge för komplexiteten som är likvärdigt när Robbie gick i spiral för att hitta muren.

Avståndet till öppning: D
N (#steg i sista vändan) ≈ 2D
Aritmetisk summa:
\[ n \times \frac{(n+1)}{2} \]
\[ 2D \times \frac{(2D+1)}{2} = D \times (2D+1) = 2D^2 + D \approx 2D^2 \]
Procedur Hitta_port
Antal_steg := 0
Om HinderF? Vrid (-1)
Upprepa ∞ ggr
{Om HinderH?=falskt
   Gå_vänster(Antal_steg)
Om HinderH? Exit Hitta_port
  Vrid (+1)
  Vrid (+1)
Om HinderV?=falskt
   Gå_höger(Antal_steg)
Om HinderH? Exit Hitta_port
  Vrid (+1)
  Vrid (+1)
}

Procedur Gå_vänster(Antal_steg)
{Antal_steg := Antal_steg + 1
Upprepa Antal_steg ggr
   Om HinderH?
      Steg
   Eljest
      Exit Gå_vänster
}

Procedur Gå_höger(Antal_steg)
*På samma sätt som Gå_vänster*