

NumberGuess

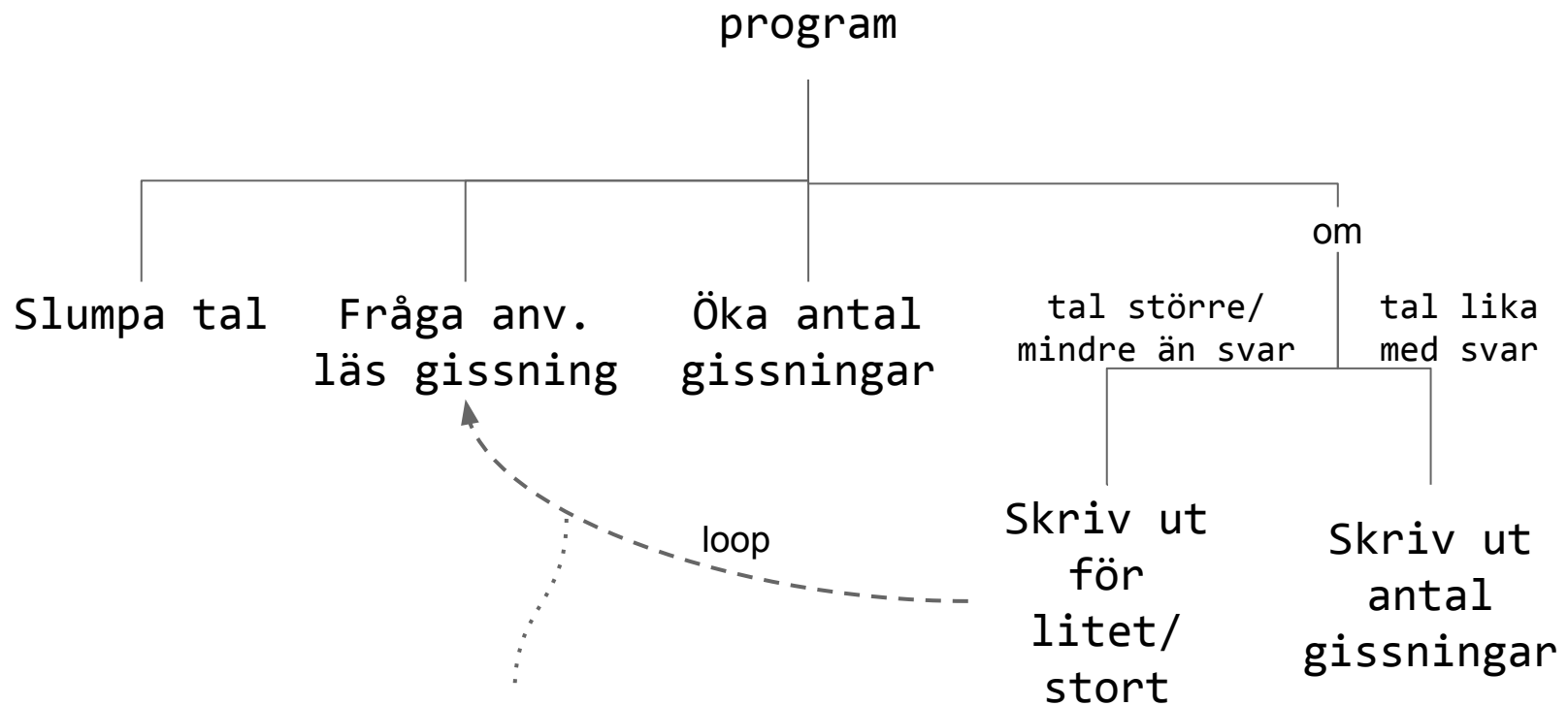
Ett program exempel

Joachim von Hacht

Ett Spel

```
Welcome to Number Guess
Guess a number [1-100] > 34
To big
Guess a number [1-100] > 22
To big
Guess a number [1-100] > 14
To small
Guess a number [1-100] > 20
To big
Guess a number [1-100] > 19
Correct! Number of guesses needed 5
```

Skiss



Kan vi göra något en gång så kan vi göra det flera gånger, upprepa med en loop!

Gör det en gång först. Testa. Lägg loop runt!

Frågor ...

- Hur kan datorn att slumpa ett tal?
- Hur kan vi avgöra olika fall; för stort, för litet eller rätt gissat?
- Hur håller vi reda på antal gissningar?
- Hur kan vi upprepa frågan?
- Hur avbryter vi om svaret är rätt?

Slumptal

Finns färdigt API för slumptal.

- M.h.a. detta kan vi skapa en slumptalsgenerator och ge den ett namn vanligen **rand**.
- M.h.a. rand kan vi fråga efter slumptal t.ex. **rand.nextInt(100)** ,ger ett tal [0-99]

Logiska uttryck

Logiska (booleska) uttryck representerar värdet true eller false.

Använder relations eller logiska operatorer

Relations: ==, !=, >, <, >=, <=

Logiska: !, &&, ||

Val (selektion)

Ett val mellan olika satser gör m.h.a. if-satsen (olika varianter på).

Parentesen efter if-satsen innehåller ett logisk uttryck

- Om sant så körs satser i if-delen
- Om falskt så körs satserna vid else-delen.

Vad som körs visas genom att vi skriver ett block vid varje del (par av { } runt satserna)

In- och dekrementering

Vi kan öka eller minska en variabel med 1 genom att använda ++ eller -- efter variabelnamnet t.ex.

```
int i = 0;  
i++;      // i blir 1
```

OBS! En variabel kan bara ändras med tilldelning eller ++/--

Uppprepning (iteration)

En upprepning kan göras m.h.a. en while-sats (loop)

Parentesen efter while-satsen innehåller ett logisk uttryck

- Om sant så körs satser i det efterföljande blocket
- Om falskt hoppas hela blocket över

Man kan hopp “direkt” ur en loop m.h.a. break-satsen

Programmet

```
final Random rand = new Random();
void program() {
    int guess;
    int nGuesses = 0;
    int theNumber;

    out.println("Welcome to Number Guess");
    theNumber = rand.nextInt(100) + 1;
    while (true) {
        // ----- Input -----
        out.print("Guess a number [1-100] > ");
        guess = sc.nextInt();
        // -- Process -----
        nGuesses++;
        // ---- Output -----
        if (guess == theNumber) {
            break;
        } else if (guess < theNumber) {
            out.println("To small ");
        } else {
            out.println("To big ");
        }
    }
    out.println("Correct! Number of guesses needed " + nGuesses);
}
```

Programmet v. 2

```
final Random rand = new Random();
void program() {
    int guess = -1;    // Ensure that guess is not equals theValue
    int nGuesses = 0;
    int theNumber;

    out.println("Welcome to Number Guess");
    theNumber = rand.nextInt(100) + 1;
    while (guess != theNumber) {                                (<---- Other expression here)
        // ----- Input -----
        out.print("Guess a number [1-100] > ");
        guess = sc.nextInt();

        // -- Process -----
        nGuesses++;

        // ---- Output -----
        if (guess < theNumber) {
            out.println("To small ");
        } else if ( guess > theNumber){    (<-- Adjust else written out for correct answer)
            out.println("To big ");
        } else {
            // Nothing
        }
    }
    out.println("Correct! Number of guesses needed " + nGuesses);
}
```

Sammanfattning

- Finns ett API för slumpstal
- Logiska uttryck representerar värdet true eller false
- Val görs m.h.a. if-satsen
- In- dekrementering görs med ++/--
- Upprepning (loop) kan göras med while-satsen
- Ett program kan skrivas på flera olika (likvärdiga) sätt