Föreläsning

Avgörbarhet och oavgörbarhet

Oavgörbara problem

Bevis av oavgörbarhet

Gemensam egenskap

Rekursiv och rekursivt uppräknelig

AVGÖRBART OCH OAVGÖRBART

ETT PROBLEM SOM HAR EN ALGORITM SOM FÖR ALLA INSTANSER KAN HITTA EN LÖSNING I ÄNDLIG TID KALLAS AVGÖRBART.

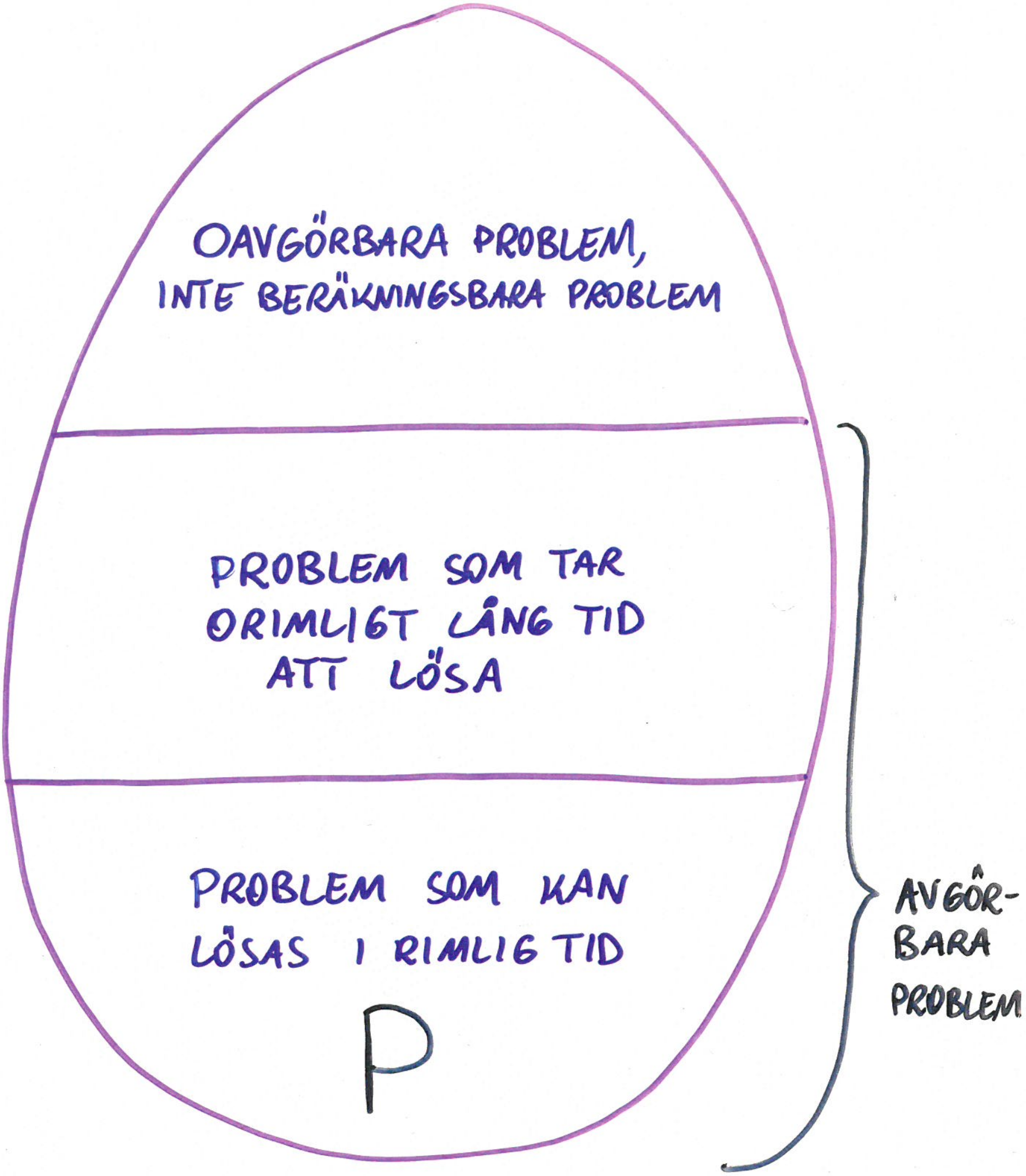
ETT PROBLEM SOM INTE KAN LÖSAS I ÄNDLIG TID AV NÅGON ALGORITM KALLAS OAVGÖRBART.

VARJE PROBLEM SOM BARA HAR ETT ÄNDLIGT ANTAL PROBLEMINSTANSER ÄR AVGÖRBART.

...FÖRUTSATT ATT VARJE LÖSNING KAN SKRIVAS NER I ÄNDLIG TID

AVGÖRBARHET/OAVGÖRBARHET ANVÄNDS I FÖRSTA HAND OM BESLUTSPROBLEM.

ANNARS TALAR MAN OM BERÄKNINGSBARHET.


OAVGÖRBARA PROBLEM,
INTE BERÄKNINGSBARA PROBLEM
PROBLEM SOM TAR ORIMLIGT LÅNG TID ATT LÖSA
PROBLEM SOM KAN LÖSAS I RIMLIG TID
P
AVGÖR-BARA PROBLEM

KAKELLÄGGARPROBLEMET

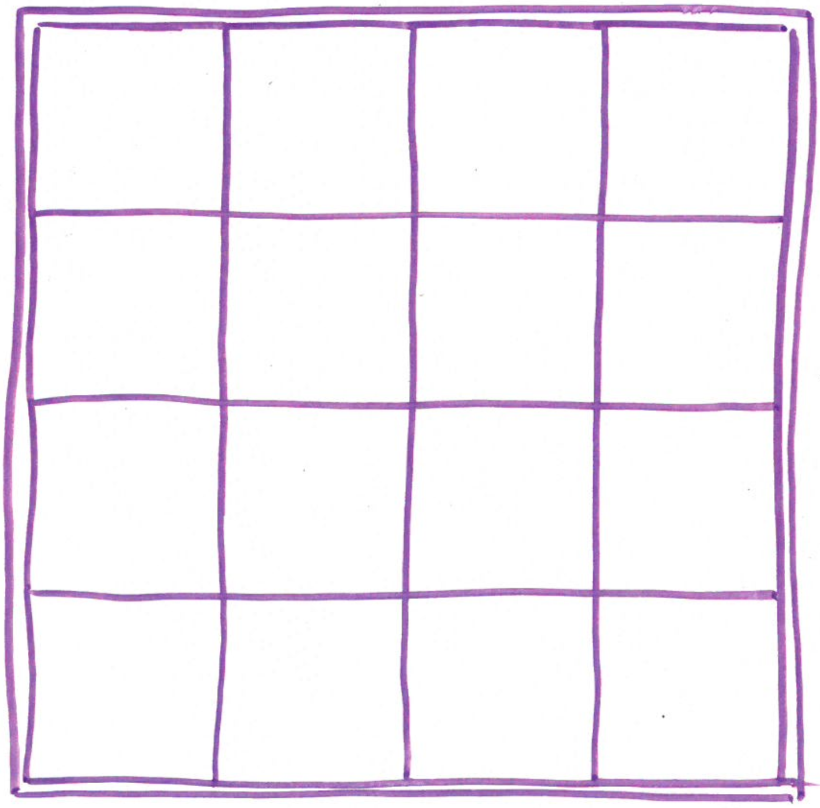
INSTANS: ETT ANTAL TYPER AV KVADRATISKA KAKELPLATTOR, DÄR EN VISS KANT SKA VARA NEDÅT.

FRÅGA: GÅR DET ATT KAKLA VARJE $m \times m$-RUTA MED KAKELPLATTOR AV ENDAST DOM GIVNA TYPERNA SÅ ATT MÖNSTRET STÄMMER ÖVERALLT?

EXEMPEL:

TYP 1 TYP 2 TYP 3

4x4:



OAVGÖRBART PROBLEM 2: ORDÖVERENSSTÄMMELSE

INSTANS: MÄNGD ORDPAR $\{(x_i, y_i)\}$

FRÅGA: FINNS DET NÅGON ÄNDLIG TALFÖLJD $a_1, a_2, \ldots$
SÅ ATT $x_{a_1} x_{a_2} x_{a_3} \cdots = y_{a_1} y_{a_2} y_{a_3} \cdots$?

EXEMPEL 1:

{ (abb, bbab), (a, aa), (bab, ab), (baba, aa), (aba, a) }
1 2 3 4 5

ÖVERENSSTÄMMELSE FÅS MED 2, 1, 1, 4, 1, 5:

2 1 1 4 1 5
a|abb|abb|baba|abb|aba
aa|bbab|bbab|aa|bbab|a
2 1 1 4 1 5

EXEMPEL 2:

{ (bb, bab), (a, aa), (bab, ab), (baba, aa), (aba, a) }

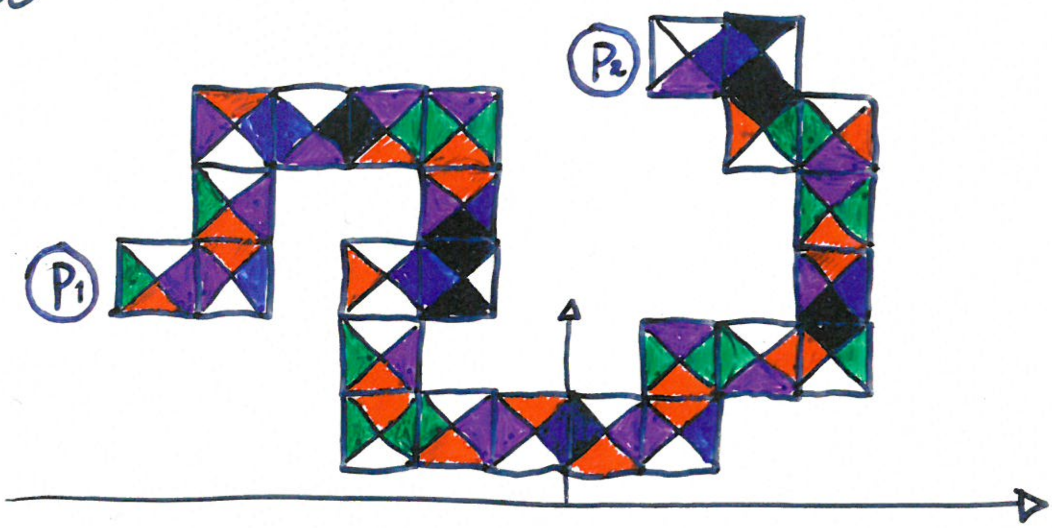
INGEN ÖVERENSSTÄMMELSE KAN FÅS!

OAVGÖRBART PROBLEM 3: ORM I KAKEL

INSTANS: MÄNGD TYPER AV KAKELPLATTOR,
TVÅ PUNKTER, p_1 OCH p_2, I PLANET.

FRÅGA: GÅR DET ATT KAKLA EN "ORM" SOM BÖRJAR I p_1 OCH SLUTAR I p_2, DÄR MÖNSTRET PASSAR ÖVERALLT OCH DÄR ORMEN HÅLLER SEJ I ÖVRE HALVPLANET?

EXEMPEL:



AVGÖRBAR VARIANT AV ORM I KAKEL:

SAMMA PROBLEM, MEN ORMEN BEHÖVER INTE HÅLLA SEJ I ÖVRE HALVPLANET.

OAVGÖRBART PROBLEM 4: STOPPROBLEMET

INSTANS: PROGRAM P OCH INMATNING X TILL PROGRAMMET

FRÅGA: KOMMER PROGRAMMET P NÅGONSIN ATT STANNA OM DET STARTAS MED INDATA X?

OAVGÖRBART PROBLEM 5: PROGRAMVERIFIERING

INSTANS: PROGRAM P OCH SPECIFIKATION S SOM BESKRIVER VAD P SKA MATA UT FÖR VARJE INDATA.

FRÅGA: UPPFYLLER PROGRAMMET SPECIFIKATIONEN?

(P UPPFYLLER SPECIFIKATIONEN OM DET FÖR VARJE MÖJLIGT INDATA STANNAR OCH PRODUCERAR DEN LÖSNING SOM ANGES AV S)

BEVIS AV OAVGÖRBARHET

REDUCERA ETT PROBLEM SOM DU VET ÄR OAVGÖRBART TILL DITT PROBLEM.

OM REDUKTIONEN I SEJ ÄR BERÄKNINGSBAR SÅ ÄR OCKSÅ DITT PROBLEM OAVGÖRBART.

EXEMPEL: VISA ATT DETTA PROBLEM ÄR OAVGÖRBART:

INSTANS: PROGRAM P
FRÅGA: STANNAR P PÅ ALLA INDATA?

REDUKTION FRÅN STOPPROBLEMET:

```
STOPP(P, X) =
    KONSTRUERA PROGRAMMET Q(Y)=
        IF X=Y THEN P(X)
        ELSE STOP
    RETURN STOPP-FÖR-ALLA(Q)
```

STOPP-PÅ-BLANKT (P)

ÄR PROBLEMET: "STANNAR P NÅGONSIN OM DET STARTAS MED TOMT INDATA?"

VISA ATT STOPP-PÅ-BLANKT ÄR OAVGÖRBART GENOM ATT REDUCERA VANLIGA STOPPROBLEMET!

BEVIS AV STOPPROBLEMETS OAVGÖRBARHET

ANTA ATT STOPP(P,X) LÖSER STOPPROBLEMET, DVS ATT STOPPROBLEMET ÄR AVGÖRBART, OCH VISA ATT DETTA LEDER TILL EN MOTSÄGELSE.

KONSTRUERA PROGRAMMET META(P) PÅ FÖLJANDE SÄTT:

```
META (P) =
   IF STOPP(P,P) THEN
      WHILE TRUE DO BEGIN END
   ELSE RETURN
```

VAD HÄNDER VID ANROPET META(META) ?

MÖJLIGHET 1: META(META) GÅR I OÄNDLIG SLINGA;
DÅ ÄR STOPP(META,META) FALSKT OCH PROGRAMMET STANNAR. ORIMLIGT!

MÖJLIGHET 2: META(META) STANNAR SÅ SMÅNINGOM;
DÅ ÄR STOPP(META,META) SANT OCH PROGRAMMET GÅR IN I OÄNDLIG SLINGA. ORIMLIGT!

GEMENSAM EGENSKAP:

ÄNDLIG VERIFIKATION

EN LÖSNING TILL ETT NP-FULLSTÄNDIGT PROBLEM KAN VERIFIERAS I POLYNOMISK TID.

PÅ MOTSVARANDE SÄTT:

EN LÖSNING TILL NÅGOT AV DOM BESKRIVNA OAVGÖRBARA PROBLEMEN KAN VERIFIERAS I <u>ÄNDLIG</u> TID.

EXEMPEL: STOPPROBLEMET

OM P STANNAR PÅ INDATA X SÅ KAN BERÄKNINGEN VERIFIERAS I ÄNDLIG TID.

EXEMPEL: ORDÖVERENSSTÄMMELSE

OM DET GÅR ATT FÅ ÖVERENSSTÄMMELSE MED EN VISS TALFÖLJD SÅ ÄR DET LÄTT ATT KOLLA ATT

$x_{a_1} x_{a_2} \ldots = y_{a_1}, y_{a_2}, \ldots$

EXEMPEL: KAKELLÄGGNING

OM DET FINNS NÅGON $m \times m$-RUTA SOM <u>INTE</u> KAN KAKLAS SÅ KAN DET (GANSKA) LÄTT VERIFIERAS: KOLLA ALLA TÄNKBARA KAKLINGAR AV DENNA RUTA.

DUBBEL ÄNDLIG VERIFIKATION

FÖR KAKELLÄGGNING KAN NEJ-LÖSNINGAR VERIFIERAS.
FÖR ORDÖVERENSSTÄMMELSE KAN JA-LÖSNINGAR VERIFIERAS.

> OM MAN FÖR ETT PROBLEM BÅDE KAN VERIFIERA JA-LÖSNINGAR OCH NEJ-LÖSNINGAR SÅ MÅSTE PROBLEMET VARA AVGÖRBART.

BEVIS:

GÅ IGENOM ALLA TÄNKBARA LÖSNINGAR I ORDNING (BÖRJA MED ALLA LÖSNINGAR AV LÄNGD 1, SEDAN LÄNGD 2 OSV) OCH TESTA FÖR VAR OCH EN OM DEN ÄR EN JA-LÖSNING ELLER NEJ-LÖSNING. AVBRYT SÅ FORT EN VERIFIERING LYCKAS.

DENNA PROCEDUR ÄR ÄNDLIG EFTERSOM MAN FÖRR ELLER SENARE TESTAR EN RIKTIG LÖSNING.

Rekursiv och rekursivt uppräknelig

Rekursiv function = beräkningsbar function =

= function som stannar för alla indata.

Rekursiv mängd = språk som kan kännas igen av någon rekursiv funktion.

Rekursivt uppräknelig funktion =

= function med ändlig verifikation av ja-lösningar.

Rekursivt uppräknelig mängd = språk som kan kännas igen av någon rekursivt uppräknelig function.

R.E.-fullständiga problem = dom svåraste problemen som kan lösas med rekursivt uppräkneliga functioner.

Stopproblemet är R.E.-fullständigt