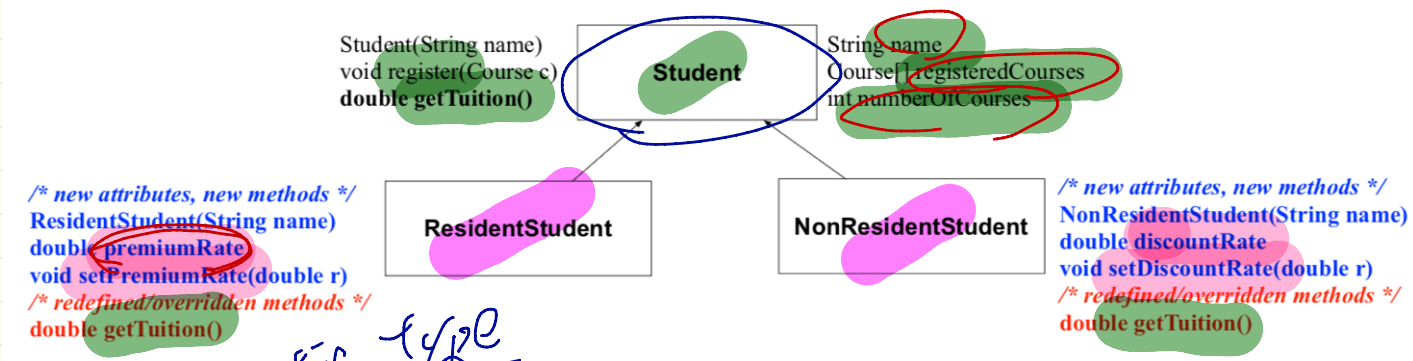
Lecture 21

Tuesday Nov. 21

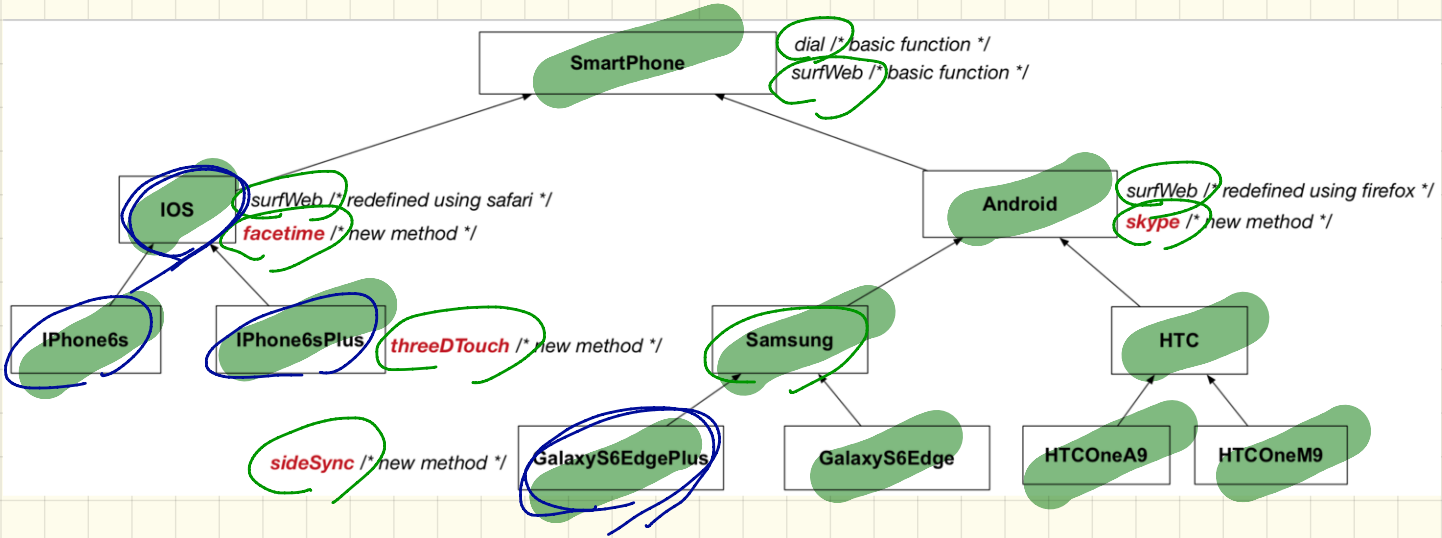


static type

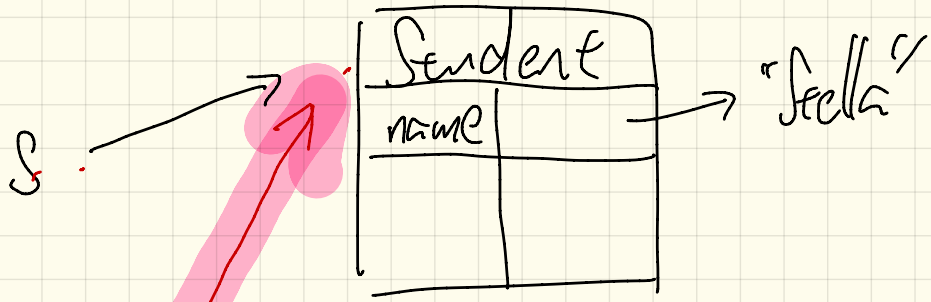
```
Student s = new Student("Stella");
ResidentStudent rs = new ResidentStudent("Rachael");
NonResidentStudent nrs = new NonResidentStudent("Nancy");
```

	name	rcs	noc	reg	getT	pr	setPR	dr	setDR
s.			✓				×		
rs.			✓				✓		×
nrs.			✓				×		✓

expectations


SmartPhone
dial /* basic function */
surfWeb /* basic function */
IOS
surfWeb /* redefined using safari */
facetime /* new method */
Android
surfWeb /* redefined using firefox */
skype /* new method */
IPhone6s
IPhone6sPlus
threeDTouch /* new method */
Samsung
HTC
sideSync /* new method */
GalaxyS6EdgePlus
GalaxyS6Edge
HTCOneA9
HTCOneM9

IOS myPhone ;



S = rS

① S = rS

② rS = S

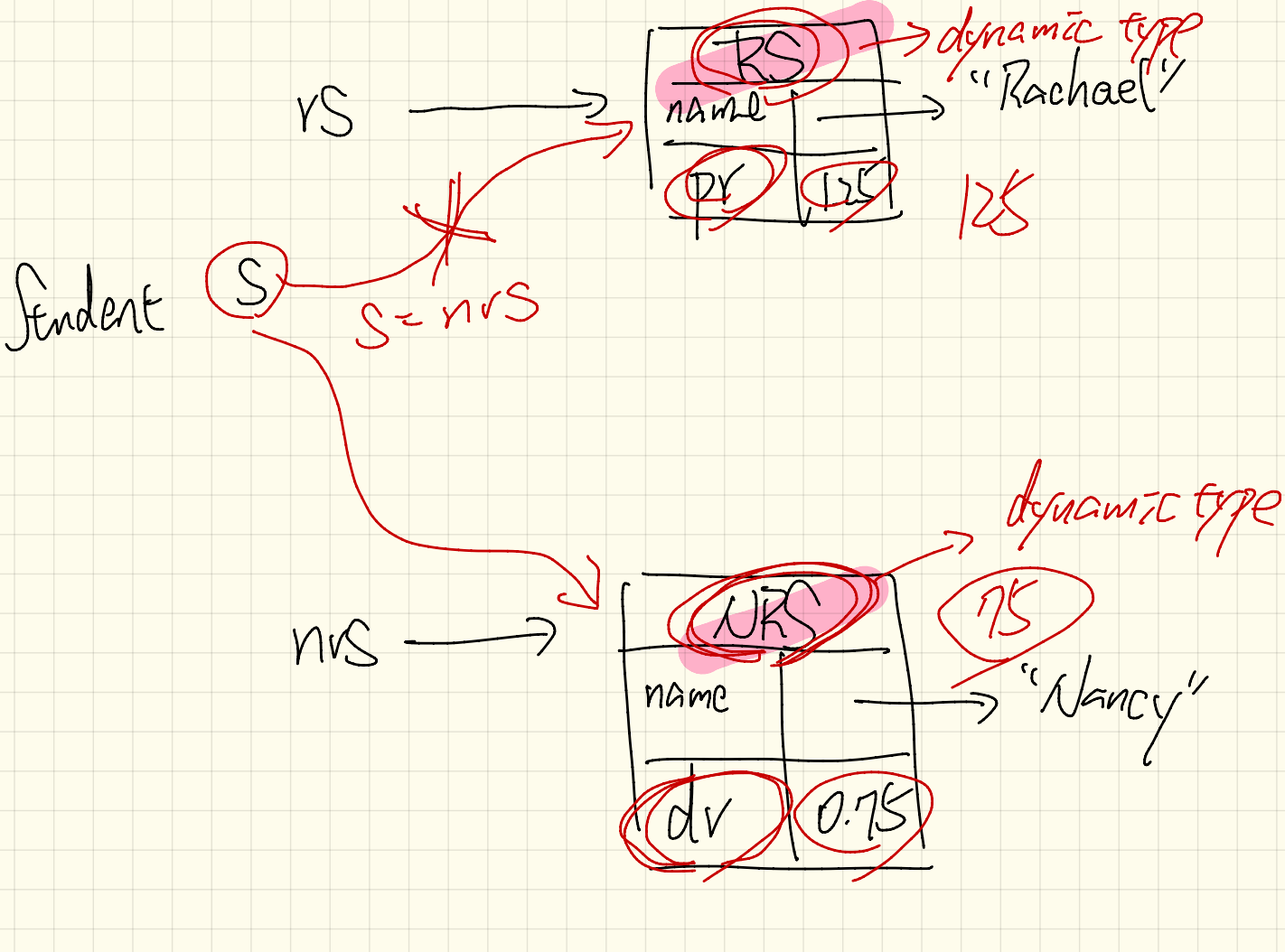
Prove by Contradiction

① Assume (rS = S) compiled

② Expectation for rS ? → Resid_Stu.

rS. premiumRate

→ crash : undefined on Stu. object.


rs
RS
name
dy
125
dynamic type
"Rachael"
125
student
s
s = nrs
nrs
NRS
name
dv
0.75
dynamic type
75
"Nancy"

Student S = new Student(...);

S → S

S.getTuition()

ResidentStudent rs = new RS(...);

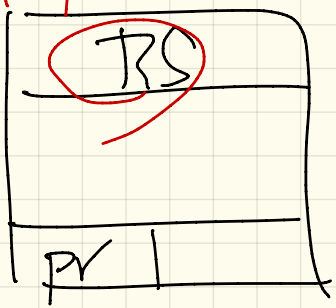
S = rs;

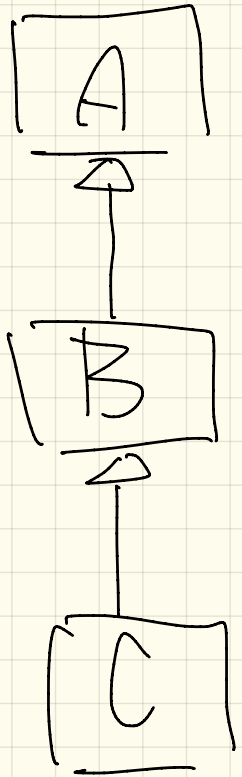
S.premiumRate ✗ ?

does not compile

Student S

ST of S (Student) does not declare premiumRate.

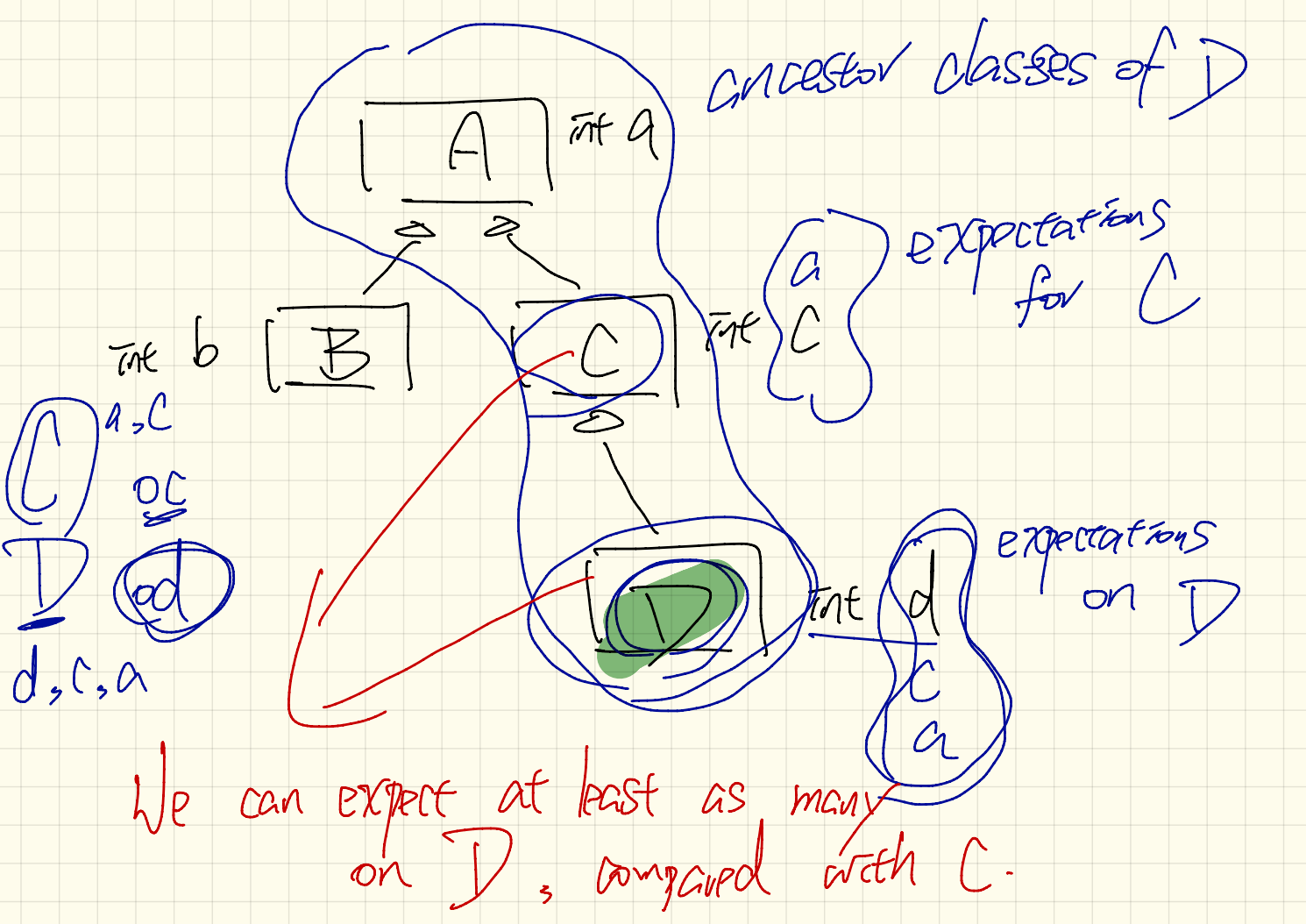




C implicitly extends A

$a > b \wedge b > C$

$a > C$

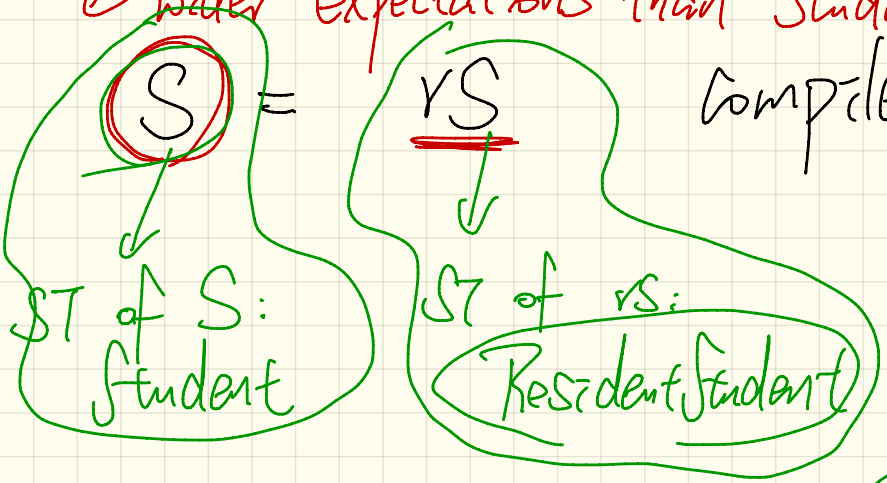

ancestor classes of D
A
int a
B
int b
C
int c
a
c
expectations for C
C
a, c
oc
D
od
d, c, a
D
int d
d
c
a
expectations on D
We can expect at least as many on D, compared with C.

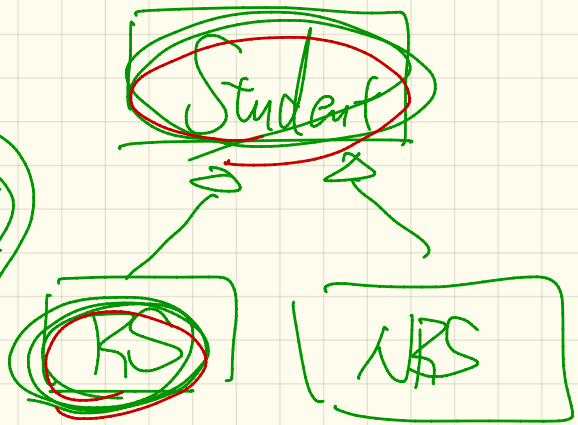
Student S = rS = S

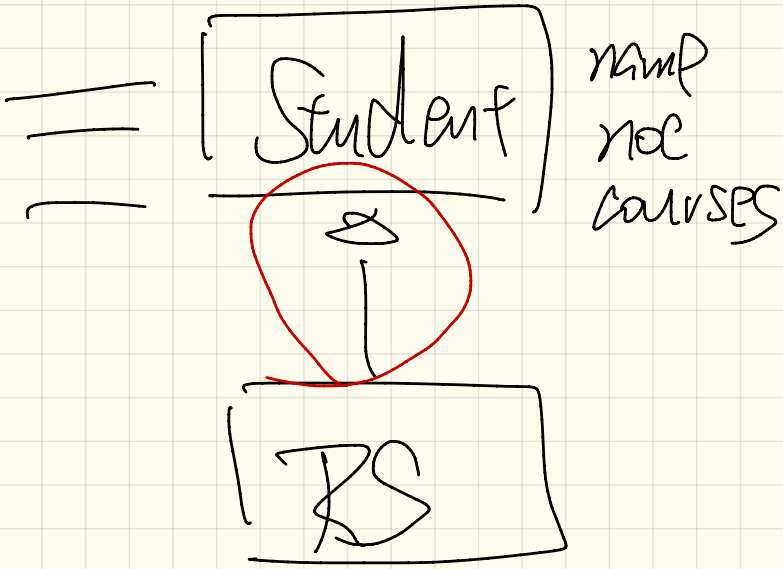
ResidentStudent rS = —

↙ wider expectations than Student (pr)

S = rS compiles ✓





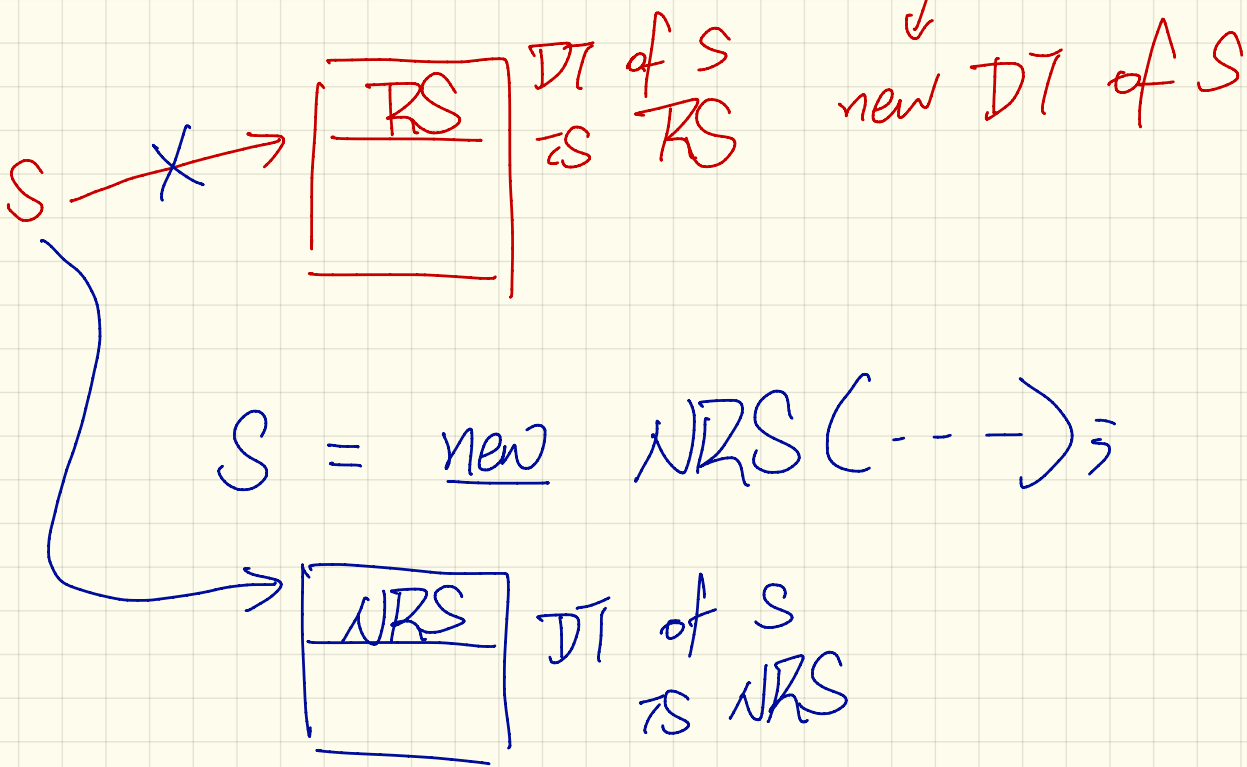


Student S = ...

ResiSet rs = ...

rs = S;

Student S = new RS(...);



① Student S = new Student();

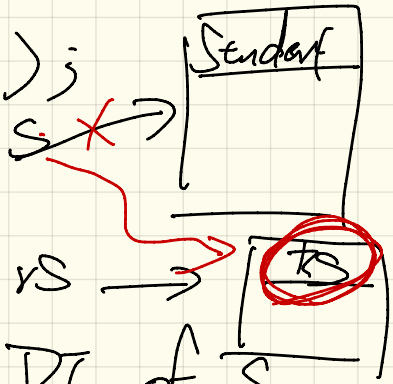
② Student rs = new RS();

③ Student nrs = new NRS();

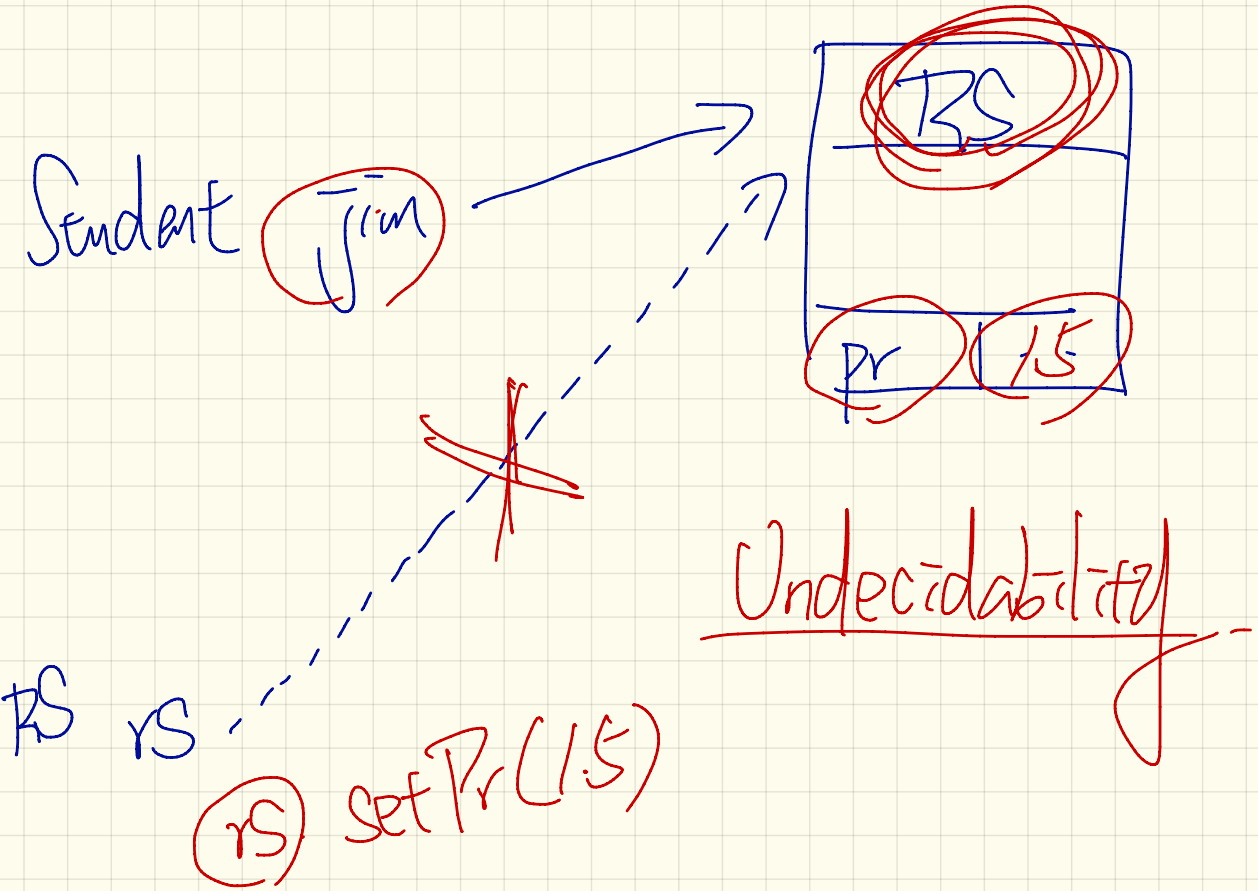
④ S = rs;

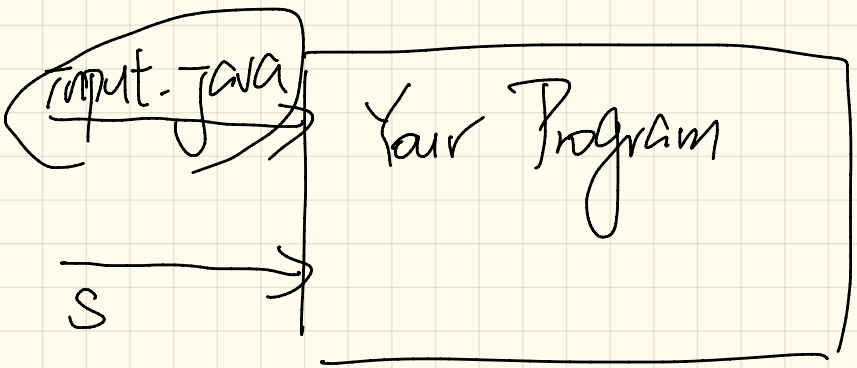
⑤ S = nrs;

ST: Student



	ST of S	DT of S	
①	Student	Student	s.getT()
②	Student	Student	
③	Student	Student	
④	Student	RS	s.getT()
⑤	Student	NRS	s.getT()


Student
Jim
RS
Pr
15
RS
rs
rs
setPr(15)
Undecidability



last DT of s
in
input.java.

input.java

```
Student s = null;
while(true){
}
s = new RS(..);
```