University of Pisa
MSc in Computer Engineering

Systems for Strategic Management and Support

LECTURES 5 AND 6

http://www.iet.unipi.it/m.cimino/pdis/

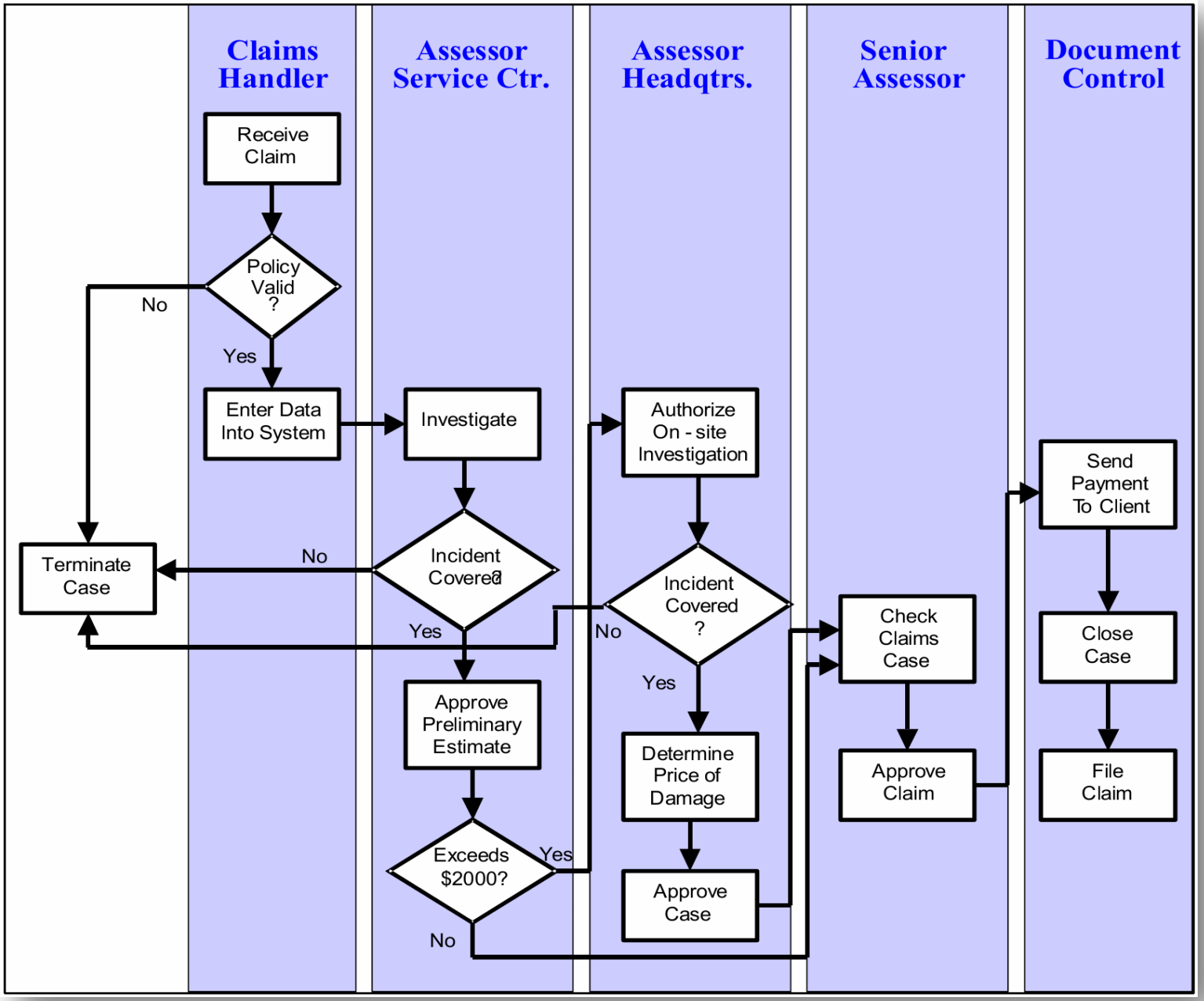
Mario G.C.A. Cimino
Department of Information Engineering

Modeling from informal diagrams: a claims process at an insurance company

a. A personal claims department in an insurance company handles claims made by their clients. The figure on the next page is a non-BPMN process map depicting the personal claims process in terms of swimlanes.

b. The first lane corresponds to work done by a claims handler (CH) located at the client's local service center. Upon arrival of a claim, the assessor determines if the client has a valid policy. If no (5% of all cases), then the case is terminated; otherwise (95% of all cases), the assessor enters the appropriate information in the system.

c. In the second lane, an assessor located at the service center (ASC) receives data from the claims handler. The assessor first determines if the claim is covered by the client's policy. If not (5% of cases), the case is terminated; otherwise (95% of cases), the assessor approves the preliminary estimate of the damage. If the damage exceeds $2,000 (35% of cases), the claim is sent to an assessor at headquarters for approval; otherwise (65% of cases), it is sent directly to a Senior Assessor.

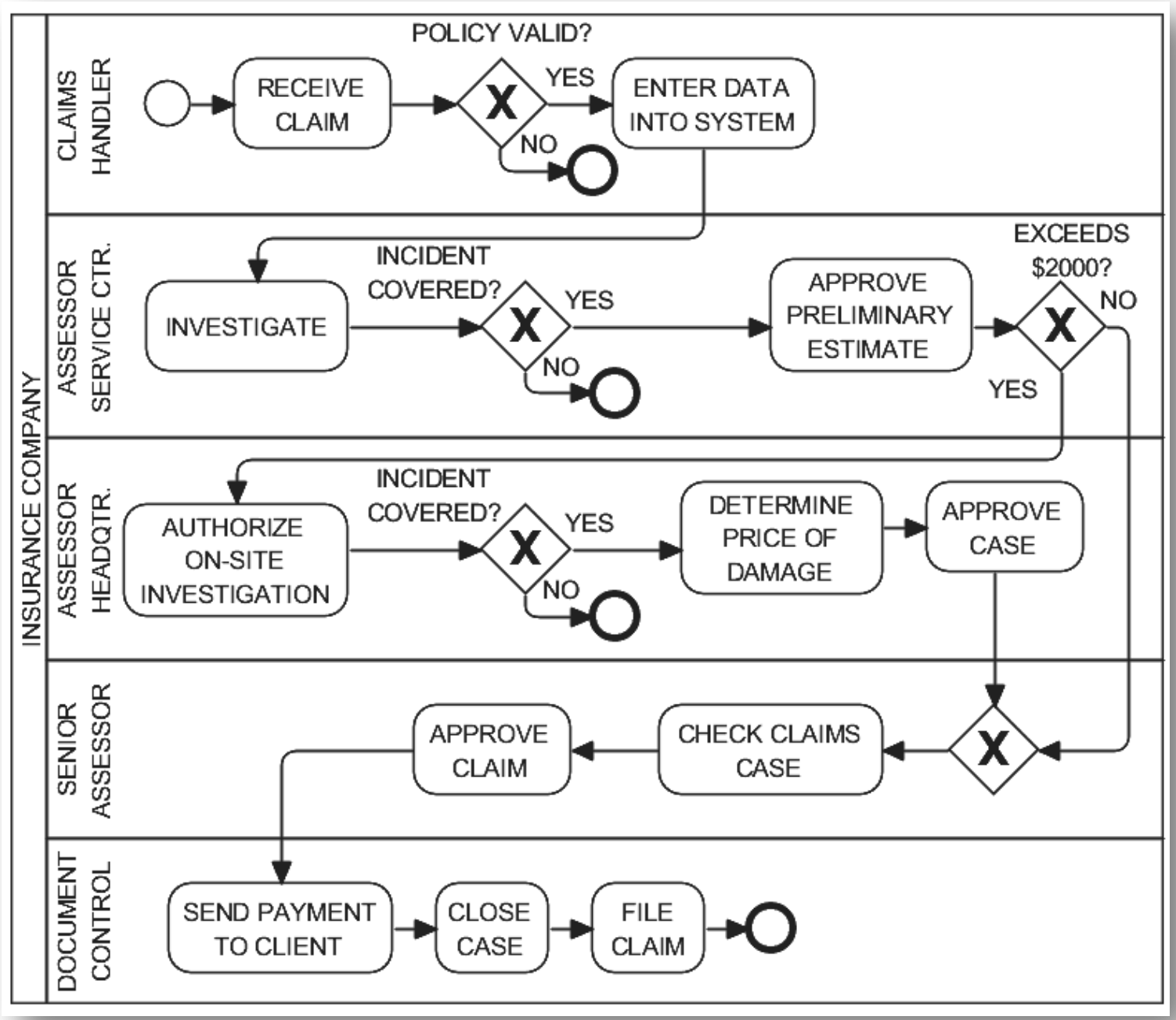

Claims Handler
Assessor Service Ctr.
Assessor Headqtrs.
Senior Assessor
Document Control
Receive Claim
Policy Valid ?
No
Yes
Enter Data Into System
Investigate
Authorize On - site Investigation
Send Payment To Client
Terminate Case
No
Incident Covered?
Incident Covered ?
Check Claims Case
Close Case
Yes
No
Yes
Approve Preliminary Estimate
Determine Price of Damage
Approve Claim
File Claim
Exceeds $2000?
Yes
Approve Case
No

d. Lane 3 corresponds to the assessor at headquarters (AHQ). The assessor first authorizes the on-site investigation of the accident. If the investigation determines that the incident is not covered by the client's policy (2% of cases), then the case is terminated; otherwise (98% of cases), a final price is determined and the case is approved.

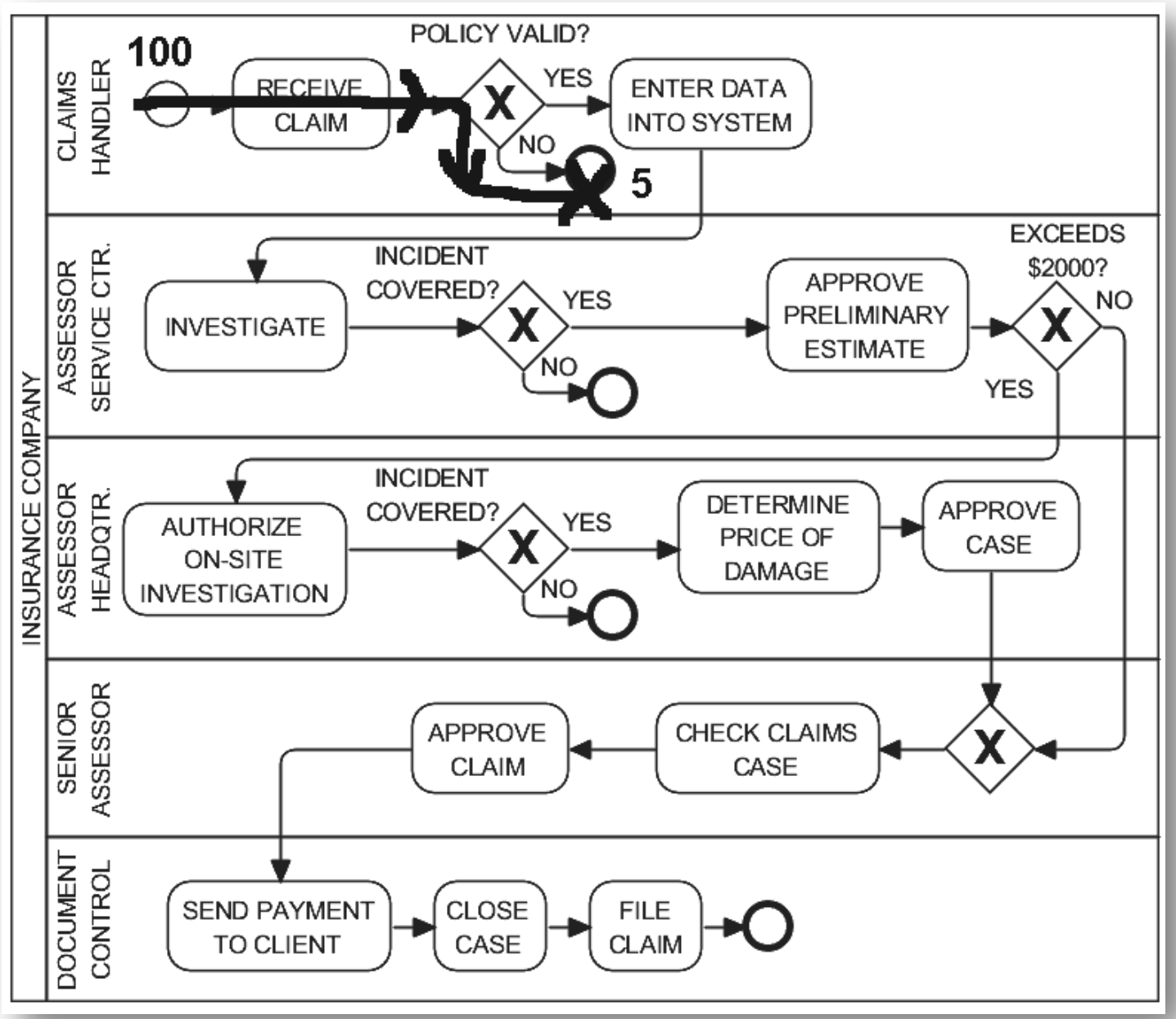
e. In lane 4, the senior assessor (SA) receives the claim, checks it, completes it, and provides the final approval. Once the claim is approved, it is sent to documentation control.

f. Documentation control (DC), in lane 5, is in charge of processing the payment to the client, closing the case, and, finally, filing the claim.

- ❑ Create a BPMN model
- ❑ Given 100 starting tokens, determine the number of ending tokens for each scenario (path), considering the aforementioned branching proportion (percentage of cases) for each gateway.


INSURANCE COMPANY
CLAIMS HANDLER
POLICY VALID?
RECEIVE CLAIM
YES
NO
ENTER DATA INTO SYSTEM
ASSESSOR SERVICE CTR.
INVESTIGATE
INCIDENT COVERED?
YES
NO
APPROVE PRELIMINARY ESTIMATE
EXCEEDS $2000?
NO
YES
ASSESSOR HEADQTR.
AUTHORIZE ON-SITE INVESTIGATION
INCIDENT COVERED?
YES
NO
DETERMINE PRICE OF DAMAGE
APPROVE CASE
SENIOR ASSESSOR
CHECK CLAIMS CASE
APPROVE CLAIM
DOCUMENT CONTROL
SEND PAYMENT TO CLIENT
CLOSE CASE
FILE CLAIM

S1)
POLICY NOT VALID:
100 x .05 = **5**

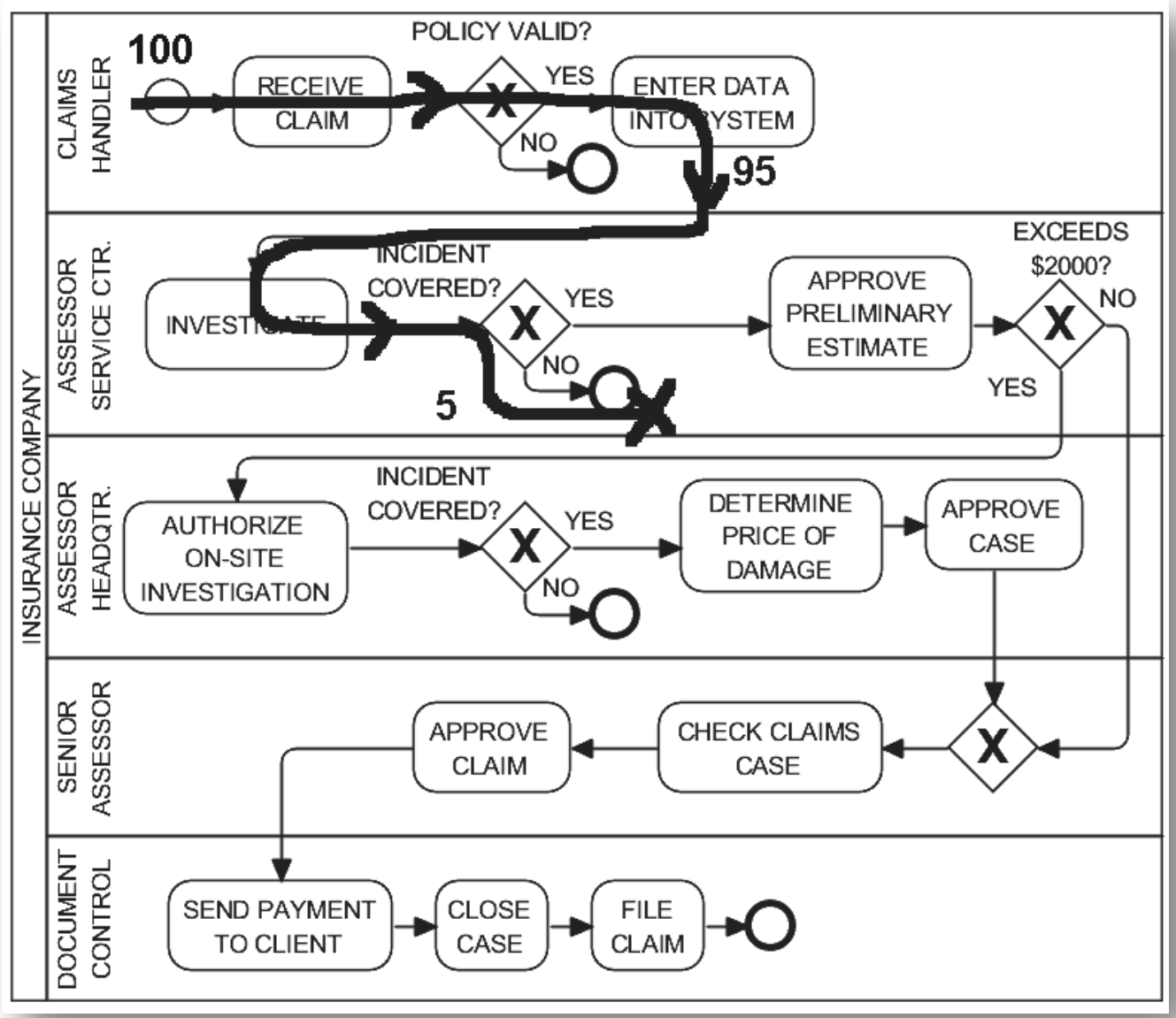


S2)
POLICY VALID &
INCIDENT NOT COV.:
100 x .95 &
95 x .05 = **5**

Note that the exact result is $4.75 \approx 5$.

Given a number of tokens arriving to a decision node:

(i) apply the percentage covered by the decision variable
(ii) add the operator "&"
(iii) report the next number of tokens rounded to the nearest integer.



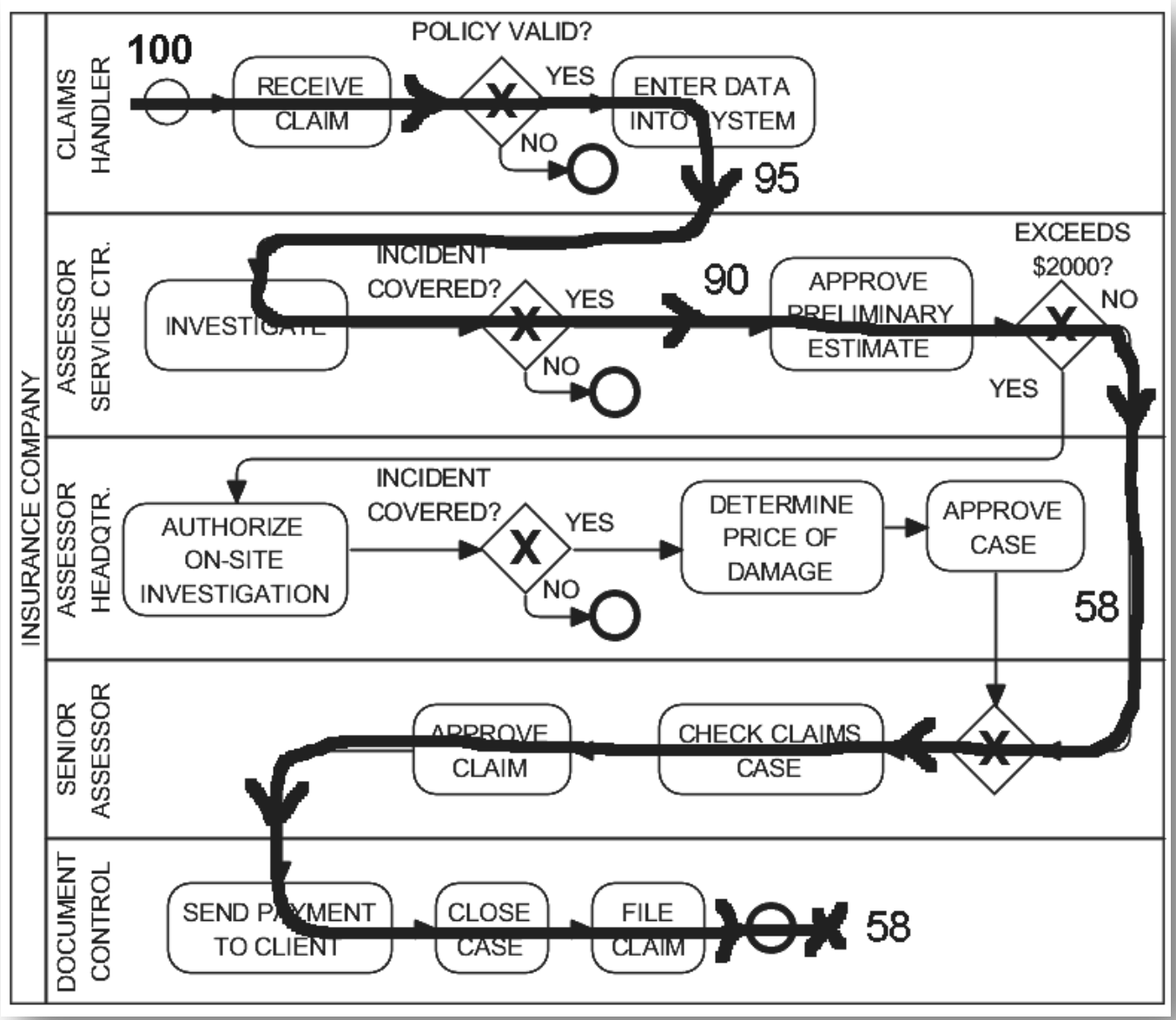
S3)
POLICY VALID &
INCIDENT COV. &
NOT EXCEEDS 2K:
100 x .95 &
95 x .95 &
90 x .65 = **58**

Note that the exact result is 58.5 ≈ 58.

If the decimal part is .5, you can round-up | round-down, thus adding | removing a half token.

But in the opposite scenario you have to make the opposite choice, removing | adding the half token.

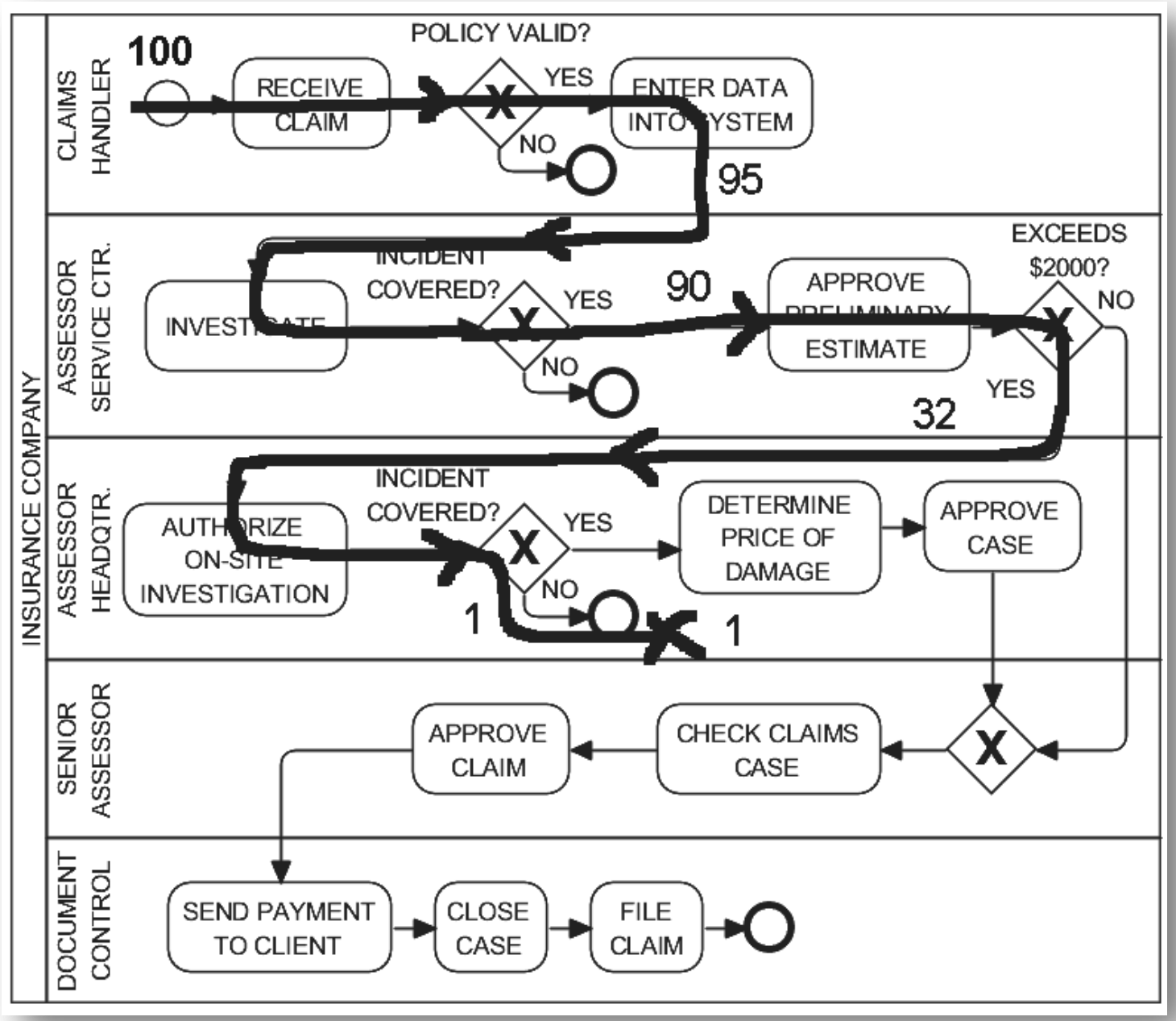
This to avoid removing | adding a half token in both scenarios.



In brief: include the half-token in a unique scenario, to avoid a whole token disappears/appears.

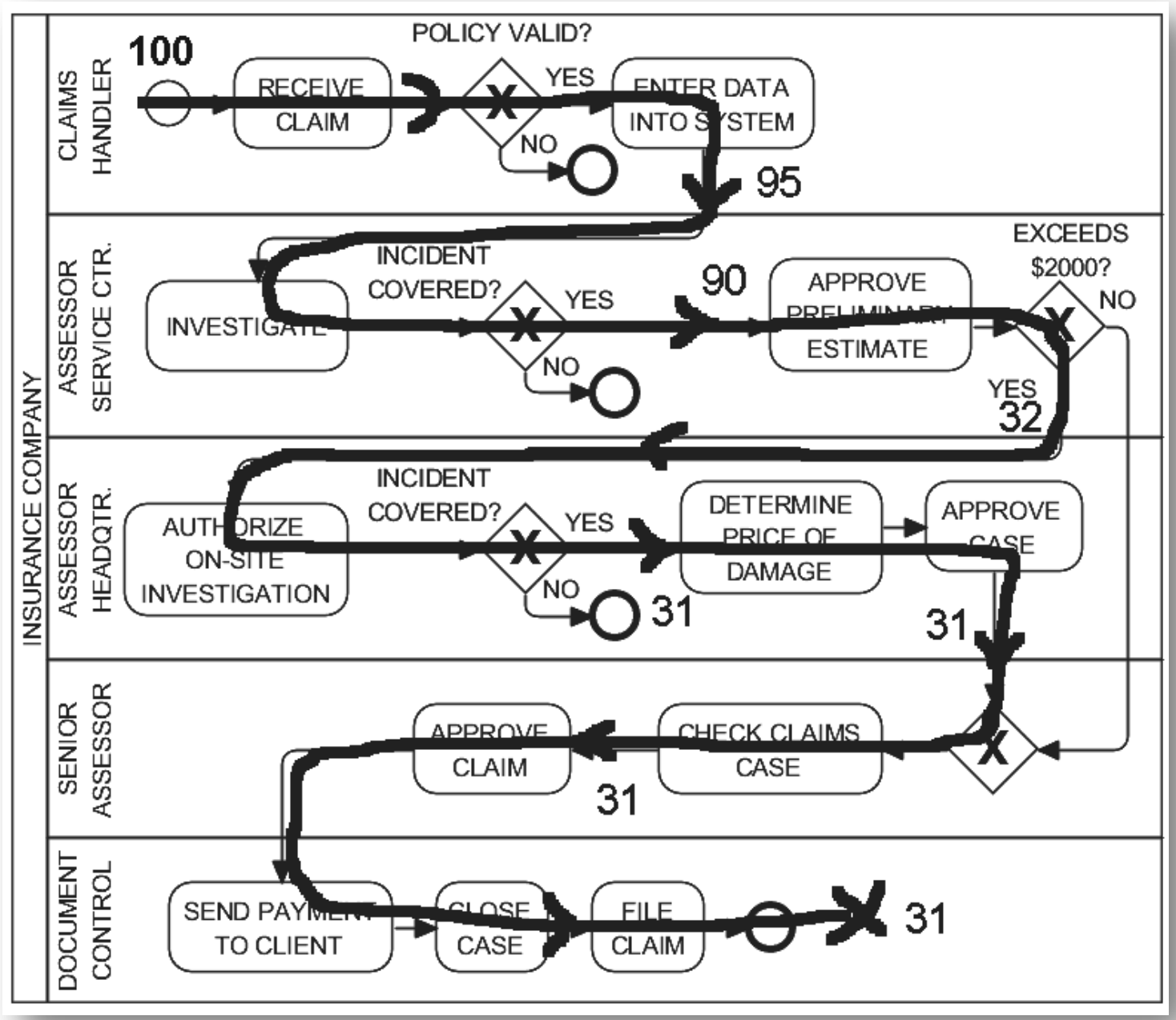
S4)
POLICY VALID &
INCIDENT COV. &
EXCEEDS 2K &
INCID. NOT COV.:
100 x .95 &
95 x .95 &
90 x .35 &
32 x .02 = **1**

Note that the exact result of 90 x .35 = 31.5 ≈ 32.



S5)
POLICY VALID &
INCIDENT COV. &
EXCEEDS 2K &
INCIDENT COV.:
100 x .95 &
95 x .95 &
90 x .35 &
32 x .98 = **31**

S1 + S2 + S3 +
S4 + S5 =
5 + 5 + 58 +
1 + 31 = **100**



Modeling from informal natural language: a Hospital Emergency Center

a. Consider the operation of a Hospital Emergency Center (HEC). The process begins when a patient arrives through the Acceptation process in the Entrance Room of the HEC, and ends when a patient is either released from the HEC or admitted into the hospital for further treatment.

b. Patients arriving on their own, after acceptation, sign in, and then are assessed in terms of their condition (triage). Depending on their condition, patients must then go through the registration process and through the treatment process before being released or admitted into the hospital.

c. Arriving patients are classified into different codes (levels), according to their condition. With Red Code (14.8% of all patients), patients are more critical than with Yellow and Green codes. Indeed, Red Code patients are taken to an Emergency Room (ER) immediately after acceptation. Once in the room, they undergo their treatment. Finally, they complete the registration process before being either released or admitted into the hospital for further treatment. Patients arriving by ambulance (5.6% of all patients) are directly classified with Red Code.

Modeling from informal natural language: a Hospital Emergency Center

d. Yellow and Green code patients must first sign in with an Administrative Clerk. After signing in, their condition is assessed by a Triage Nurse, and then they are taken to an Emergency room. Once in the room, Yellow and Green code patients must first complete their registration, then go on to receive their treatment, and, finally, they are either released or admitted into the hospital for further treatment.

e. In terms of resources, the treatment process consists of the following activities: a secondary assessment performed by a nurse and a physician; laboratory tests, performed by a patient care technician; the treatment itself, performed by a nurse and a physician.

f. In terms of resources, the registration process consists of the following activities: a data collection activity performed by an Administrative Clerk; an additional data collection activity performed by an Administrative Clerk, in case the patient has Worker's Compensation Insurance; a printing of the patient's medical chart for future reference, performed by an Administrative Clerk.

Modeling from informal natural language: a Hospital Emergency Center

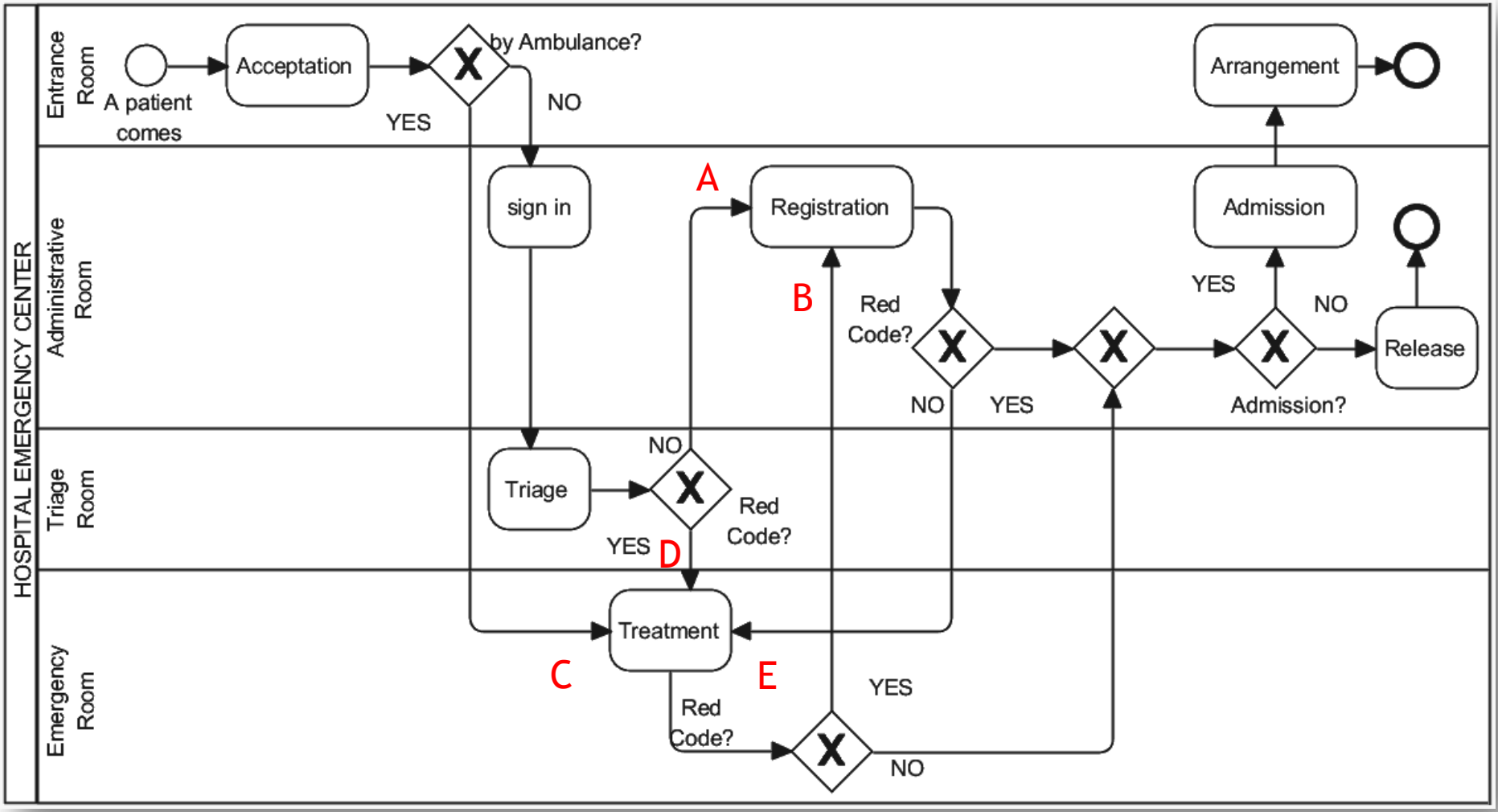
g. Finally, 90% of all patients (regardless of the assigned code) are released from the HEC, while the remaining 10% are admitted into the hospital for further treatment.

h. The final release/hospital admission process consists of the following activities: in case of release, an administrative clerk fills out the release papers; in case of admission into the hospital, an Administrative Clerk fills out the patient's admission papers.

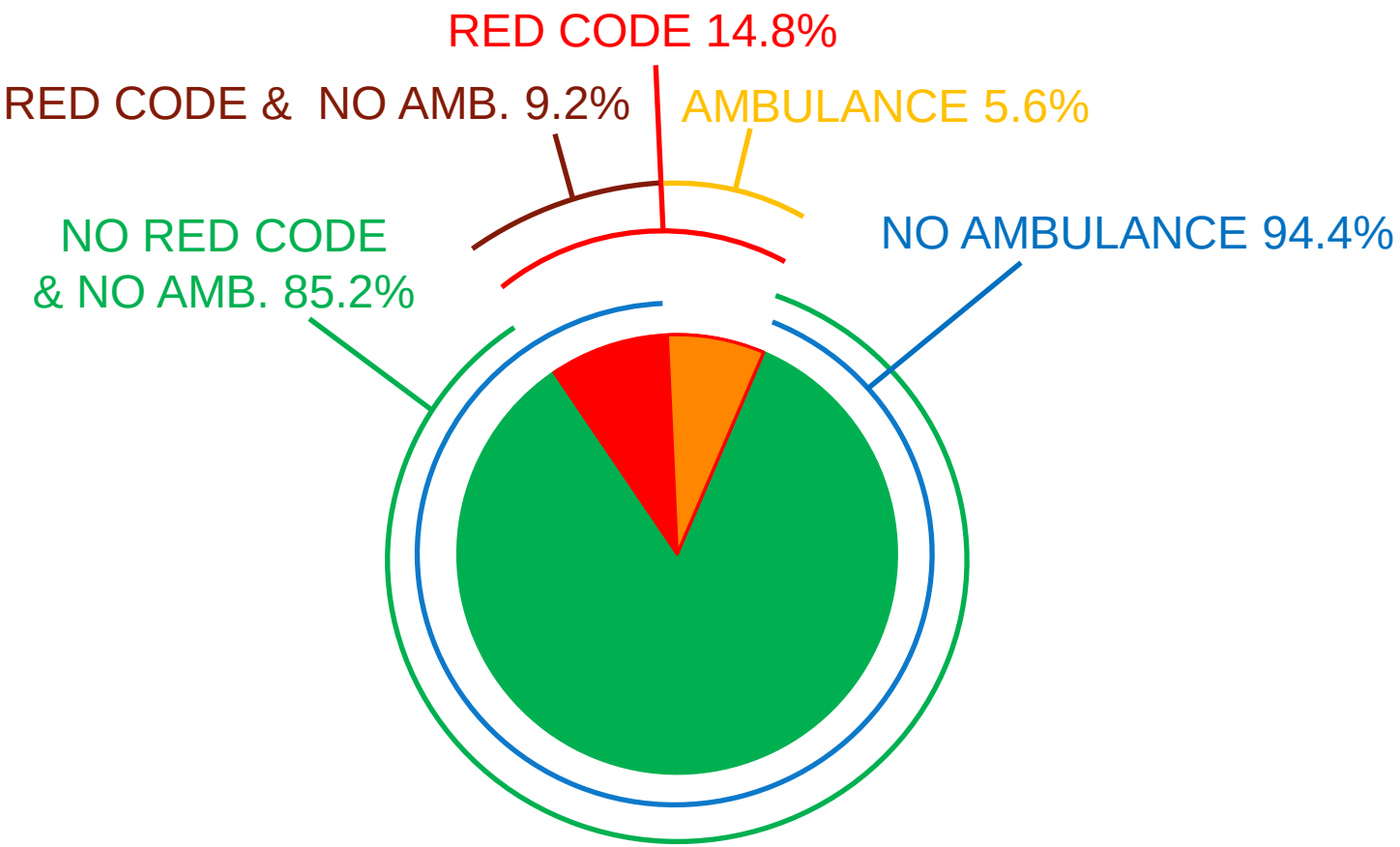
i. The patient must then wait at the Entrance Room for a hospital bed to become available (Arrangement), so as to be transferred to the hospital room.

- ❑ Create a BPMN model
- ❑ Given 100 starting tokens, determine the number of ending tokens for each scenario (path), considering the aforementioned branching proportion (percentage of cases) for each gateway: Ambulance 5.6%, Red code 14.8%, Released 90%

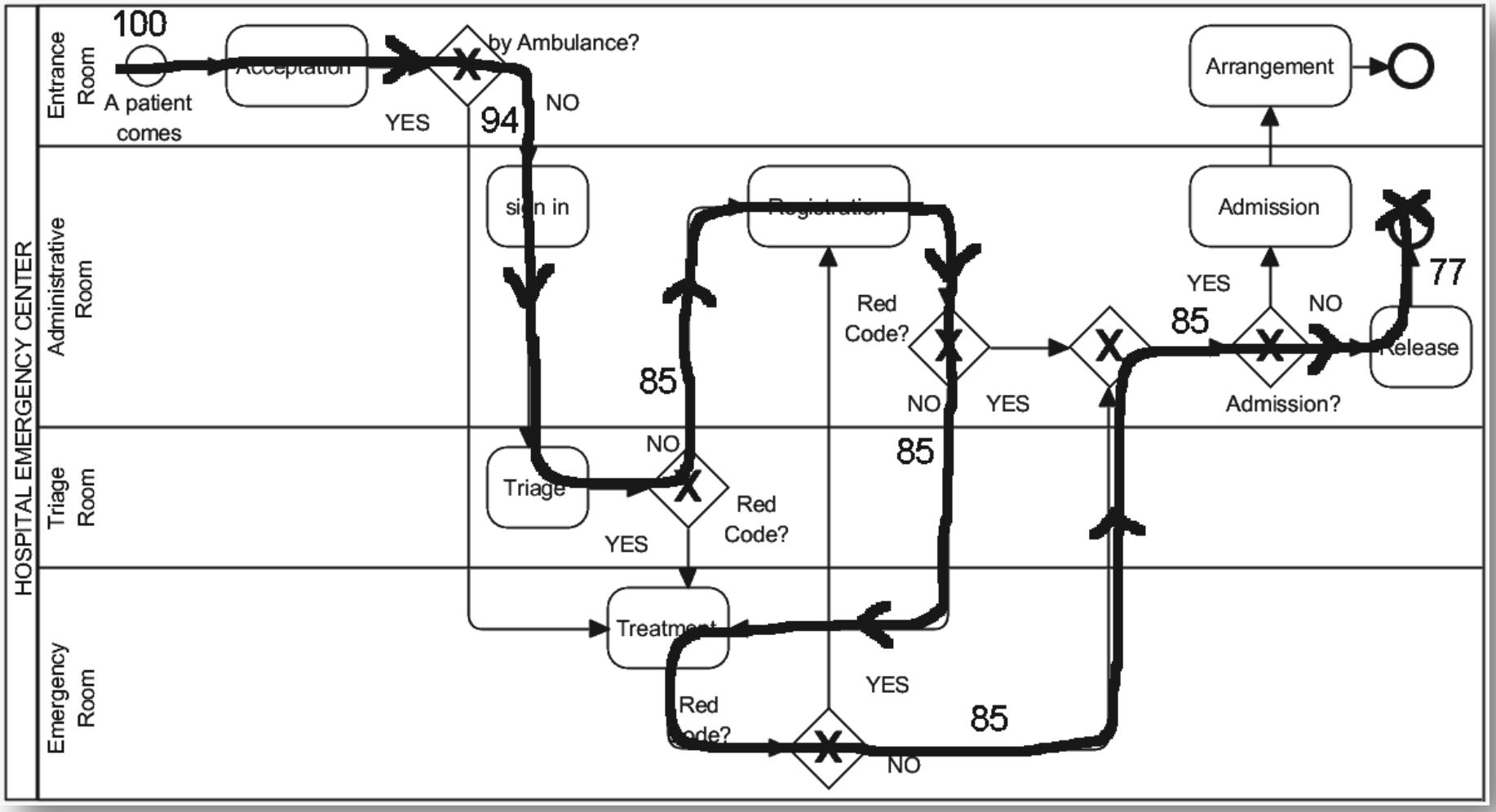
- Note that there are tasks with multiple input flows: A, B (Treatment) and C,D,E (Registration). Are they parallel inputs? Actually a XOR-join might merge them, because they are exclusive. To avoid a high number of gateways in the diagram, additional XOR-join are not inserted.
- A = *Red-Code*; B = *Not Red-Code. C = By-Ambulance;* D = *Not-By-Ambulance & Red-Code*; E = *Not-By-Ambulance & Not-Red-Code*.



The 14.8% of all patients have a red code; some of them (5.6%) come by ambulance, and then the remaining 9.2% of patients have red code and come by their own. Patients with no red code are 100% - 14.8% = 85.2%. Patients coming by their own are 100% - 5.6% = 94.4%.

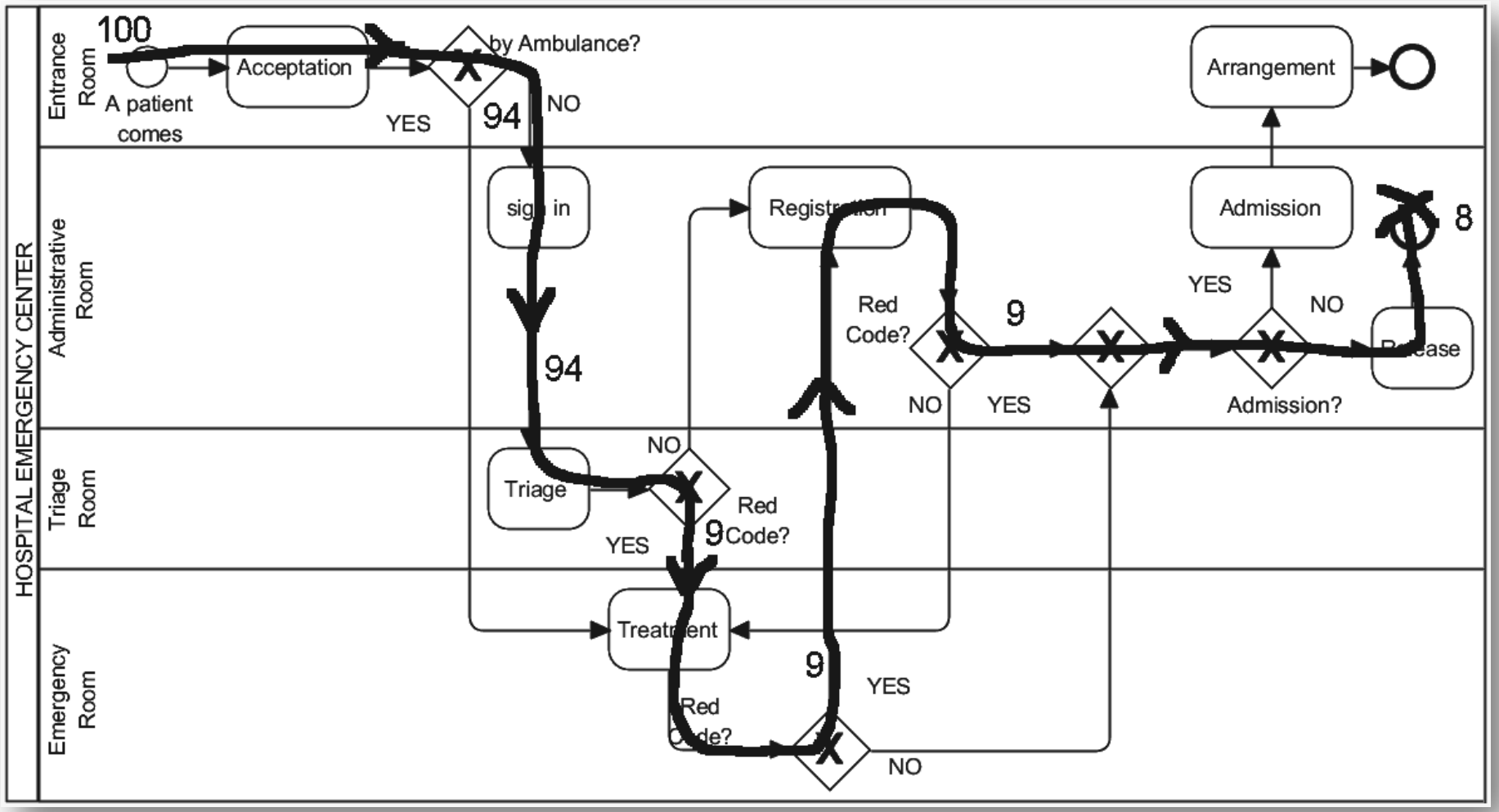


S1) NO AMBULANCE & NO RED CODE & NO ADMISSION:
100 x .944 & 100 - 14.8 & 85 x .9 = 76.5 ≈ **77**.

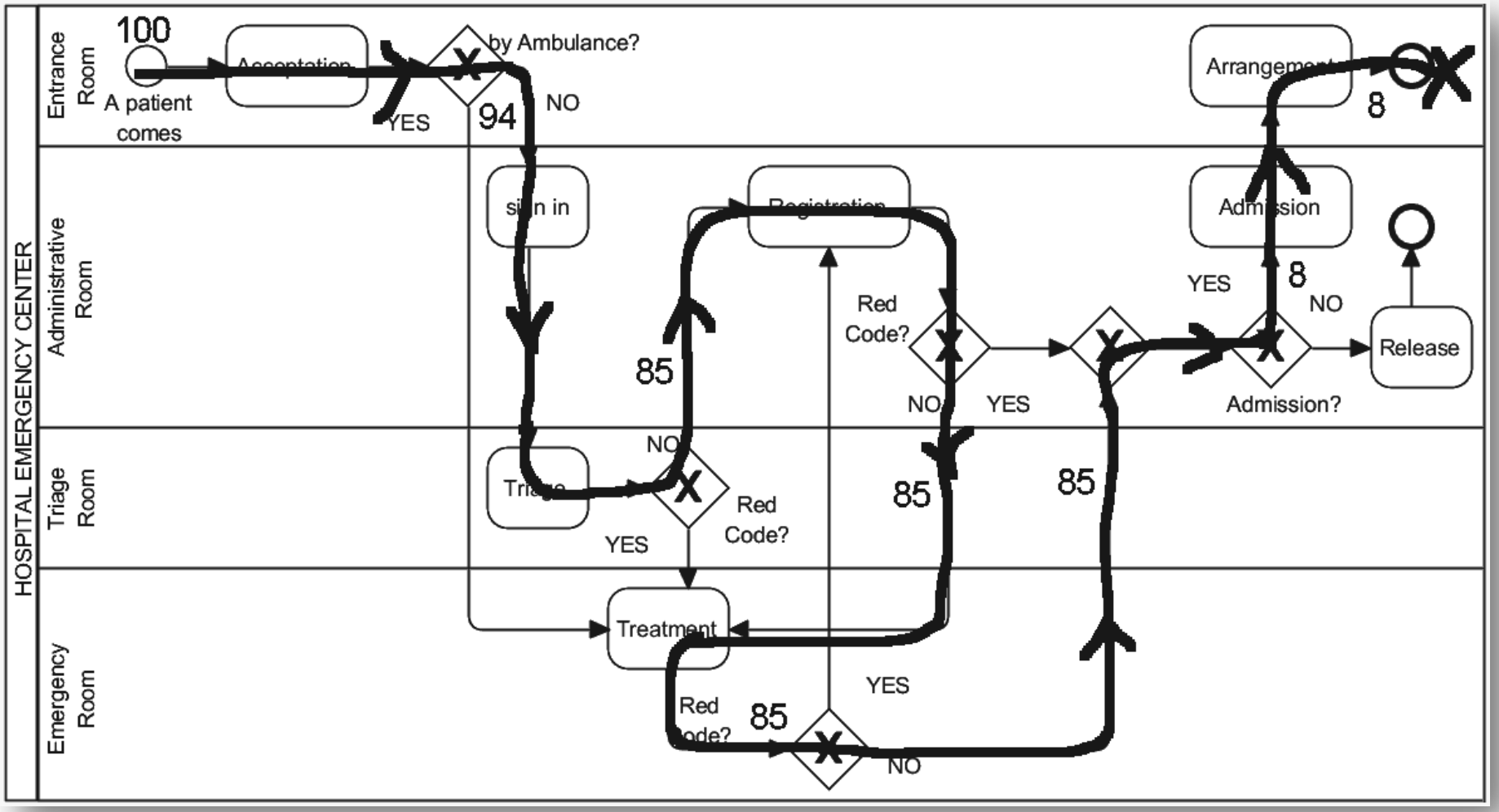


S2) NO AMBULANCE & RED CODE & NO ADMISSION:

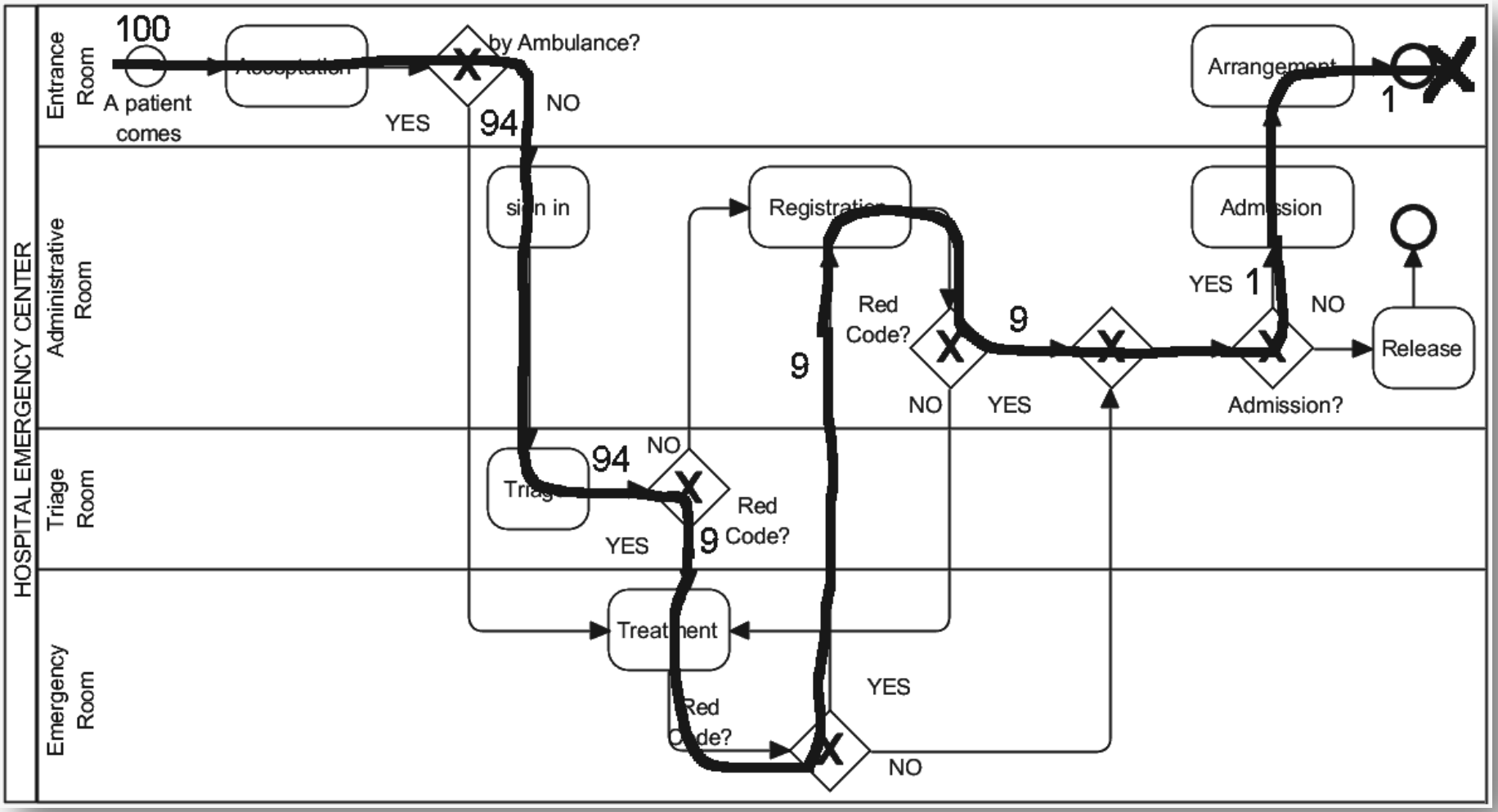
100 x .944 & 14.8 – 5.6 & 9 x .9 = **8**.



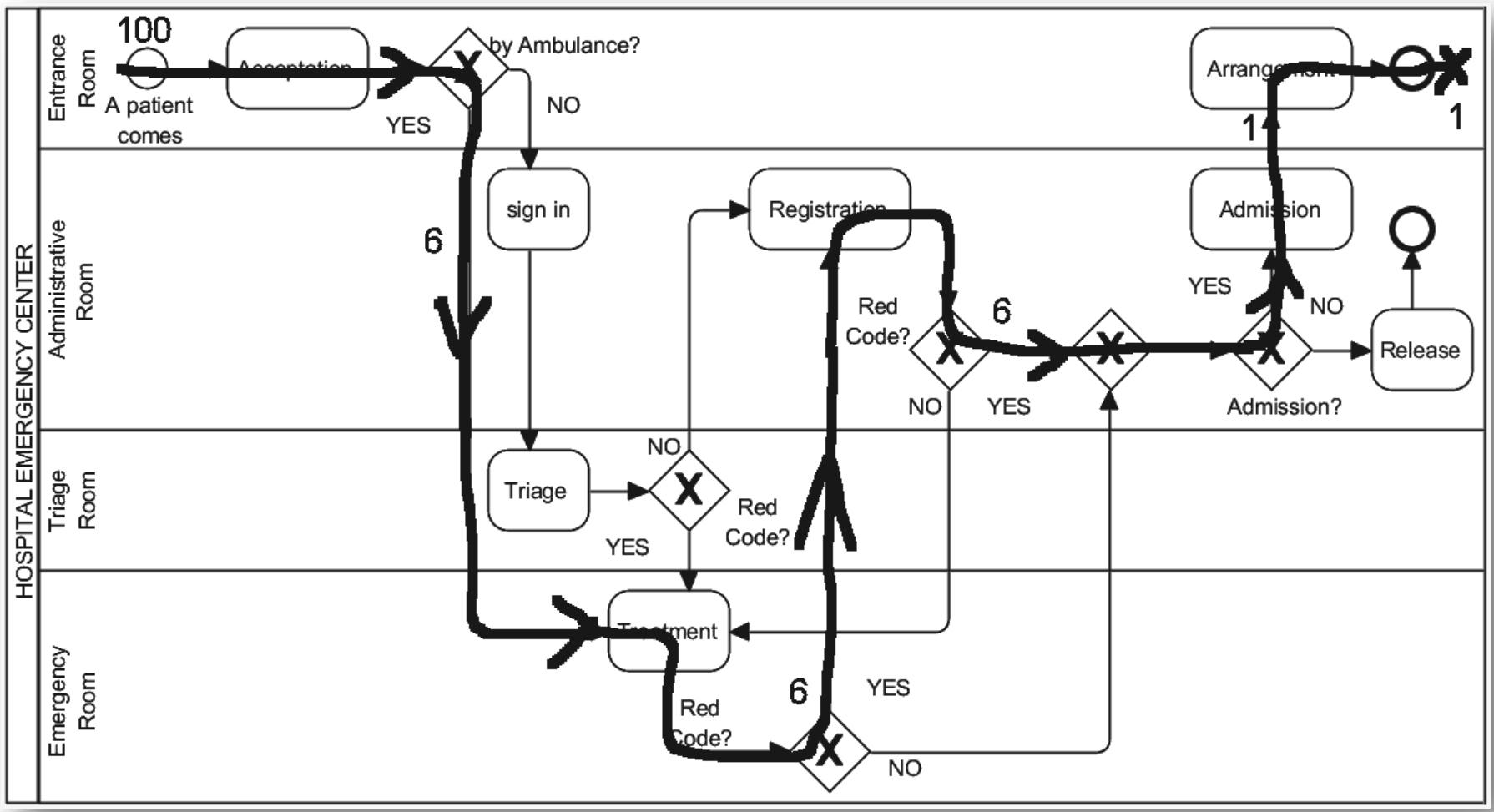
S3) NO AMBULANCE & NO RED CODE & ADMISSION:
100 X .944 & 100 - 14.8 & 85 x .1 = 8.5 ≈ **8**.



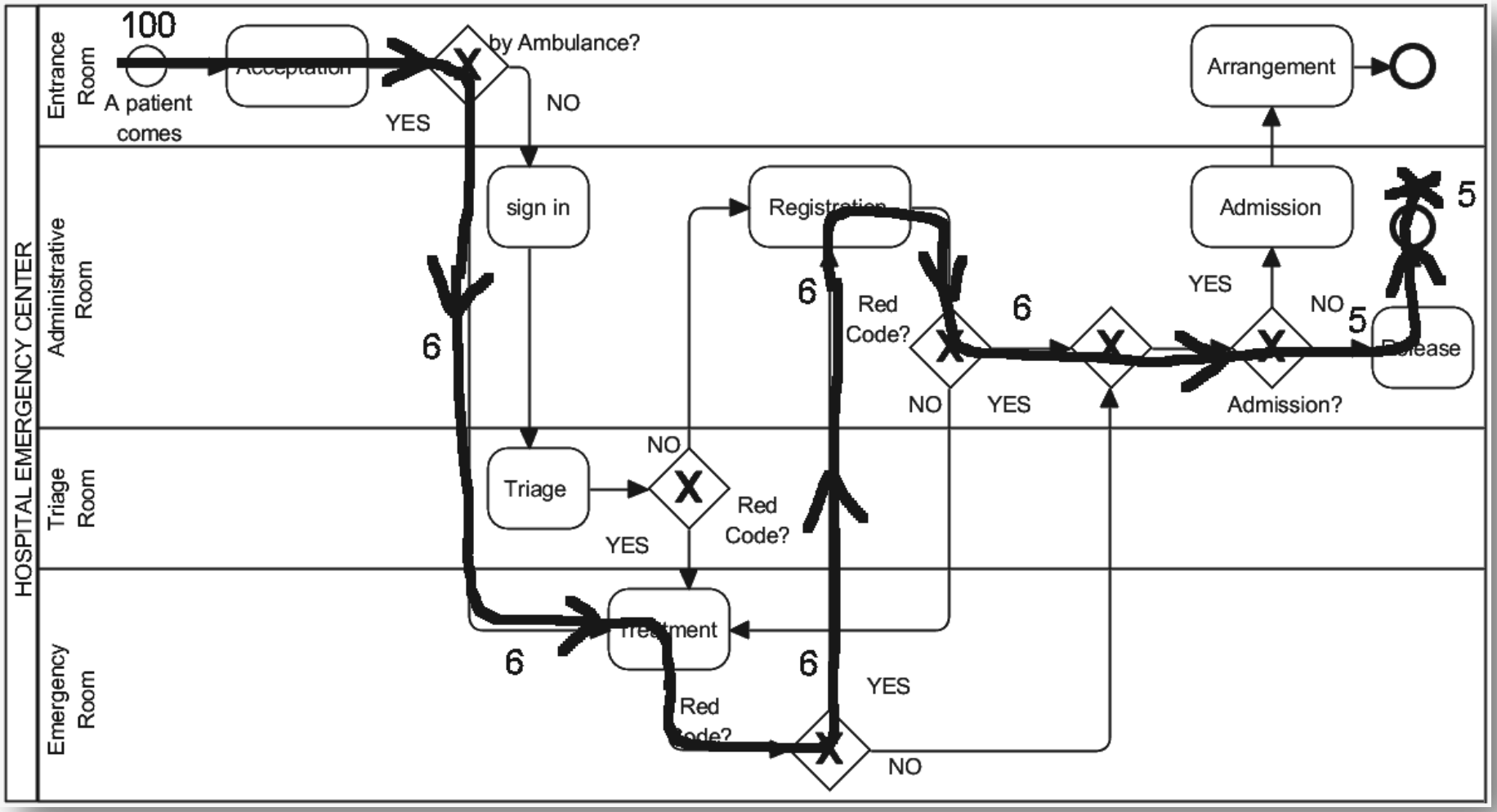
S4) NO AMBULANCE & RED CODE & ADMISSION:
100 X .944 & 14.8 – 5.6 & 9 x .1 = **1**.



S5) AMBULANCE & RED CODE & ADMISSION:
100 X .56 & 6 & 6 x .1 = **1**.

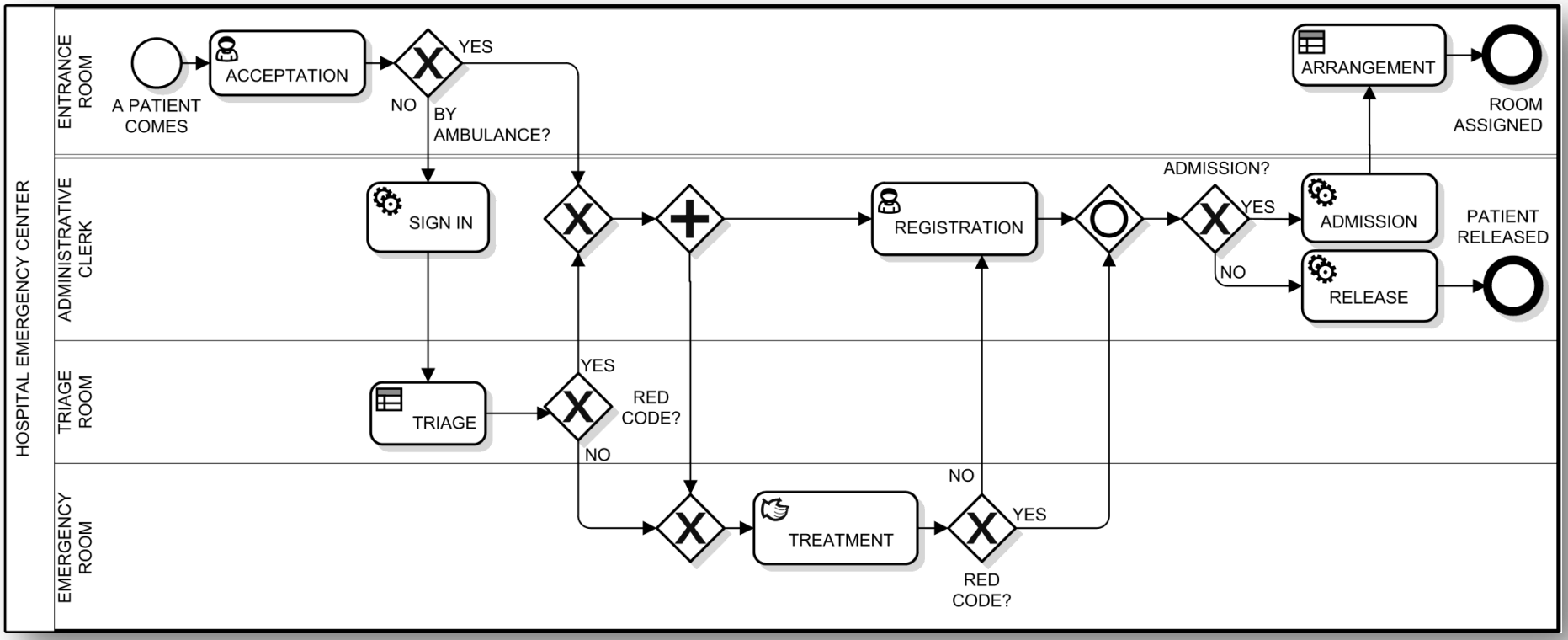


S6) AMBULANCE & RED CODE & NO ADMISSION:
100 X .56 & 6 & 6 x .9 = **5**.



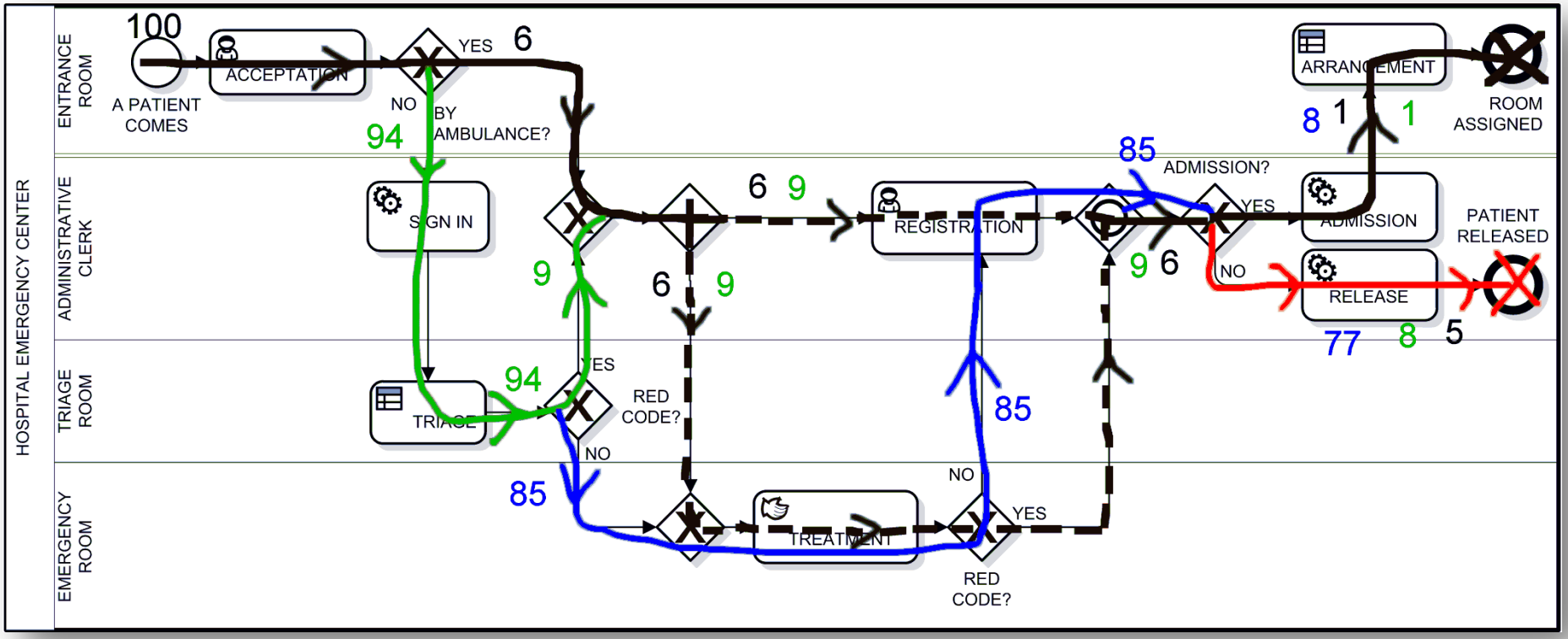
S1 + S2 + S3 + S4 + S5 + S6 = 77 + 8+ 8+ 1 + 1 + 5 = **100**

j. Redesign some features of the HEC model to improve the cycle time of patients. In the new model, we assume that patients can go through the treatment process and the registration process in parallel. That is, we assume that, while the patient is undergoing treatment, the registration process is being done by a surrogate or whoever is accompanying the patient. If the patient's condition is very critical, someone else can provide the registration data; however, if the patient's condition allows it, then the patient can provide the registration data during treatment.



- Given 100 starting tokens, determine the number of ending tokens for each scenario (path), considering the same aforementioned branching proportion.

S1) AMB & RED & ADM: 100 x .56 & 6 & 6 x .1 = **1**
S2) AMB & RED & NO ADM: 100 x .56 & 6 & 6 x .9 = **5**
S3) NO AMB & RED & ADM: 100 x .944 & 14.8 - 5.6 & 9 x .1 = **1**
S4) NO AMB & RED & NO ADM: 100 x .944 & 14.8 - 5.6 & 9 x .9 = **8**
S5) NO AMB & NO RED & ADM: 100 x .944 & 100 - 14.8 & 85 x .1 = **8**
S6) NO AMB & NO RED & NO ADM: 100 x .944 & 100 - 14.8 & 85 x .9 = **77**



Types of tasks and icons

- a *service task* is an automated function processed by an external application;
- a *receive/send* task receives/sends a message;
- a *user task* expects input by a user via a UI;
- in a *business rule task* some business rules are applied (e.g. via a business rule management system launched by the process engine) to produce a result;
- a *script task* contains statements processed directly by the process engine;
- a *manual task* is carried out without IT support;
- in an *abstract task* no type is defined.