



IS 166: *Sample Integrating Experience*

Name: _____

You have been hired by a travel company to design a system for them. When a customer contacts the company to make a reservation, the system either records (if new customer) or retrieves (if existing customer) their name and nationality. Each reservation is for a customer going on a trip. There are many possible categories of trips, each to a specific country (such as Greece), with a fixed annual starting date (May 18 of every year), and a duration (always takes 15 days). Each trip category is annual and is offered only once a year. Hence, for example, the 15-day trip to Greece is offered once a year, always on May 18. During each reservation, the agent performing the task is automatically identified by the system.

The system should be able to:

- A. Produce the Reservation-Dates Report that receives (via the telephone, from the manager requesting it) a starting date, an ending date, and a country, and then shows every reservation that was made in that period for that country in terms of the name of the agent who took the reservation, the name of the customer who made it, the reservation date, and the reservation number.
- B. Allow management to update (add/change/delete) data about customers.
- C. Allow agents to make reservations for customers.
- D. Produce the Agent-Rate Report, showing each agent and the hourly rate that person charges.
- E. Allow agents to cancel a reservation made earlier.
- F. Produce the Trip-Capacity Report, which shows each trip category and its total capacity, i.e., the maximum number of people who can be accommodated on it.
- G. Allow management to update (add/change/delete) data about various trip categories.
- H. Allow agents to modify the details of a reservation requested earlier.
- I. Allow staff, at the end of each trip, to capture/record the overall satisfaction level (1-5 scale) of each customer who was on that trip.
- J. Allow management to update (add/change/delete) data about agents.
- K. Produce the “Trip Captains” report that lets management specify a time period, and then displays the name and email address of the captain for each trip that took place (i.e., started) in that period. Each trip has one captain. Different trips to the same location have different captains every year.
- L. Allow management to update (add/change/delete) data about captains.
- M. Allow management to update (add/change/delete) data about trips.

Confine yourself to the given requirements. Do NOT expand the functionality of the system to make it “real world”. If you do so, you will lose points. This is a test of your ability to translate the GIVEN requirements into systems analysis and design models. Obviously, this system needs to be interfaced/integrated with many related subsystems, but those functions are outside the scope of this application.

1. The Database

Draw the entity-relationship diagram. Write each TABLE NAME in capital letters. Specify cardinality for each relationship by using a single arrowhead (\longrightarrow) for “one”, a double arrowhead ($\longrightarrow\!\!\!\longrightarrow$) for “many”.

Show ONLY the names of the tables; their attributes will be specified below. (1)

Define the fields (attributes) in each table. Include only those fields that are relevant to the stated requirements. Do not include any extra fields intended to portray the workings of the “real world.”

Follow these notation conventions:

- Write each TABLE NAME in capital letters.
- Underline the primary key of each table.
- Name each field in lower-case letters.
- Insert each foreign key in a box. Place all foreign keys at the end.

There is no need to produce a data dictionary unless you use attribute names that you feel are not self-explanatory. Use the following format. (2)

TABLE-NAME = primary-key, attribute 1, attribute 2, foreign-key

2. The Data Flow

A sample of the Reservation-Dates Report appears below. For this particular report, the requesting manager specified the following parameters based on the system prompting that user to do so: Starting date: October 2, 2004 ; Ending date: November 25, 2004 ; Country: Greece

Reservation Dates Report			
Greece, October 2, 2004 - November 25, 2004			
<u>Agent Name</u>	<u>Customer Name</u>	<u>Reservation Date</u>	<u>Reservation Number</u>
Peggy	John Mistro	Oct 3, 2004	R2
Bob	Gina Mastro	Nov 23, 2004	R4

Draw a data flow diagram for the process that produces this report. Show a single process. Make sure the source/destination of all data is clearly indicated, and that each arrow is labeled. Except for the name of the report itself, avoid composite labeling of data flows; i.e., each arrow label should refer to one or several specific data elements. (2)

3. The Structure

Draw an action diagram showing the overall structure of this system. The structure should be designed with maintainability in mind. More functionality of the type already discussed will be added to the system in the near future. Your structure design should anticipate these future expansions and accommodate them. Use the diamond symbol as appropriate. Avoid using the hat symbol. (6)

5. The User Interface

The Reservation-Dates Report parameters (starting date, ending date, country) are entered through a telephone.

The logic for the module that allows the manager to enter the parameters is shown below. *Sample user interaction is shown in italics.* Critique this process ONLY from the viewpoint of user-interface design; do not critique it from the standpoint of structured English design. Identify four major flaws in it by **circling** each, **numbering** each circle, and for each (on the page following it)

- explain what is wrong with it,
- suggest a solution in plain English.

You do not need to express your solutions in formal structured English modifications, only in plain English. Write your answers on the page following next. Do not comment on the quality/correctness of the structured English provided. If you identify more than four flaws, you will not receive extra points.

(4)

BEGIN

PRODUCE “Welcome to the automated report system for the Reservation-Dates Report. Please enter the month of the starting date using the numeric pad on your phone” TO user

The guest presses 10

ACCEPT start-month FROM user

*Sample user interaction
is shown in italics.*

PRODUCE “Please enter the day of the starting date using the numeric pad on your phone” TO user

The guest presses 1

ACCEPT start-day FROM user

PRODUCE “Please enter the year of the starting date using the numeric pad on your phone” TO user

The guest presses 2004

ACCEPT start-year FROM user

Starting-date = start-month/start-day/start-year (*10/01/2004*)

PRODUCE “Please enter the day of the ending date using the numeric pad on your phone” TO user

The guest presses 30

ACCEPT end-day FROM user

PRODUCE “Please enter the month of the ending date using the numeric pad on your phone” TO user

The guest presses 11

ACCEPT end-month FROM user

PRODUCE “Please enter the year of the ending date using the numeric pad on your phone” TO user

The guest presses 2004

ACCEPT end-year FROM user

Ending-date = end-month/end-day/end-year (*11/30/2004*)

PRODUCE “Please enter the name of the country using the letters on the numeric pad of your phone” TO user

The guest presses G-R-E-E-C-E

ACCEPT country FROM user (*Greece*)

Country = country

PRODUCE “Thank you for using the automated report system. The report you requested will be sent to your computer momentarily” TO user

END

5. The User Interface (Continued)

Flaw 1

Explanation:

Suggested Solution:

Flaw 2

Explanation:

Suggested Solution:

Flaw 3

Explanation:

Suggested Solution:

Flaw 4

Explanation:

Suggested Solution:

Feedback

On a scale of 1 (very bad) to 5 (very good), what is your opinion of this examination as a tool for measuring how much you have learned in this course? _____

What specific comments do you have about this examination? Your comments, whether positive or negative, will not affect your grade on this exam in any way.

INPUT

From an external entity: ACCEPT <name of data> FROM <name of external entity>
 From a previous process: GET <name of data> FROM <name of process>
 From a data store:

- Search for a matching record:
 - FIND RECORD IN <file name> WITH MATCHING <field name> IN <source>
 - FIND RECORD IN <file name> WHERE <file name>.<field name> = <source>.<field name>
- Get the value of a field in a specific record:
 - READ <field name> FROM <file name>

PROCESSING: Control Structures

Sequence	Selection: IF	Selection: CASE	Iteration
<action 1> <action 2>	IF <condition> THEN <action 1> ELSE <action 2> ENDIF <hr style="width: 50%; margin-left: 0;"/> IF <condition> THEN <action> ENDIF	DOCASE CASE <variable = value 1> <action 1> CASE <variable = value 2> <action 2> CASE <variable = value n> <action n> ENDCASE	DOWHILE <condition> <action> ENDDO <hr style="width: 50%; margin-left: 0;"/> DOUNTIL <condition> <action> ENDDO <hr style="width: 50%; margin-left: 0;"/> DOFOR <condition> <action> ENDDO

- Special case: EXIT → suspend all further processing; end the program

PROCESSING: Operations

Logical Operations

- Comparison (= > < ≥ ≤ ≠)
- Logical operators (NOT, AND, OR)

Mathematical Operations

- Assignment
 - X = X + Y : Add contents of Y to X and store the result in X
 - X = k : Store constant k in X
- Calculation (+ - * / ** ...)

OUTPUT

To an external entity: PRODUCE <name of data> TO <name of external entity>
 To another process: SEND <name of data> TO <name of process>
 To a data store:

- Creating a new record: APPEND <name of data> TO <file name>
- Updating an existing record: UPDATE <file name>: <field name(s)>