

<u>SURNAME:</u>		<u>NAME:</u>		<u>B1</u>
<u>Student ID:</u>				
<u>Teacher:</u>				

Question 1	Results
<p>Given the following number in base 10, report the number in binary, octal and hexadecimal (21)₁₀</p> <p>Given the following number in base 2, report the number in hexadecimal (11010111)₂</p>	<p>(?)₂ = _____</p> <p>(?)₈ = _____</p> <p>(?)₁₆ = _____</p> <p>(?)₁₆ = _____</p>
<p><i>Report the steps for the results</i></p>	

Question 2	Results (report the steps)
<p>Given a 32KByte memory containing the following data structure</p> <pre>struct element list[X];</pre> <p>where the struct element is defined as</p> <pre>struct element{ char n[60]; int n; float p; }</pre> <p>Determine the maximum value of X in a computer in which integers and floats are represented in 4 bytes.</p>	

Question 3
<p>Explain the difference between a source file, an object file and an executable file. Also explain what relationship exists between these different types of files.</p>

Question 4 (Programming)

Implement a C program that receives the name of a file as command line argument. The file contains, the rooms availability for the hotels in a city. For every line, the name of a hotel, the area in the city where the hotel is located and the price of the rooms in the following format:

<hotel_name> <area> <price>

where the hotel name is a single string of maximum 40 characters not including spaces, the area is a single string with one of the following possible values: **north**, **south**, **east** and **west**, and the price is a real number.

The number of lines in the file is not known a priori. The number of hotels for each area is maximum 10 (so the number of hotels is maximum 40).

The format of the file is always correct. The hotels in the file are initially considered all available to receive bookings.

The program must also be able to receive user reservation requests from the keyboard. The requests must be managed by the program in the form of an iterative menu that repeatedly asks the user to enter their request until the user enters "exit" request represented by the number 3.

The list of requests the user can make are the following:

1. Book a hotel for one night (see details below)
2. Print on the screen the list of hotels still available including areas and prices (see details below)
3. Exit

Details:

- Request 1: the program should receive also from the user the hotel area, (ex.: *1 south*) as shown in the example below. The program should book a hotel room for the user in **the requested area** choosing the hotel with **the lowest price**. Then, the program should print on the screen the name of the hotel and room price (with two decimal numbers). At this point, the booking process is done for that hotel that should be no more available from this point on. **Attention:** each hotel could be only booked once.

When there are no available hotels in the area requested by user, the program should print "**There is no available hotels in the area requested**".

- Request 2: no further information is required from the user. The program should print the list of all available hotels (see the example below).

<p>Example file: "Hotel.txt"</p> <pre>Dolomiti north 120.00 Miramare south 80.00 Gino east 60.00 Fonti north 132.00 Imperiale east 200.00 Locanda south 83.00 Siberiano north 140.00</pre>	<p>Example of program execution: c:\> reservation.exe Hotel.txt</p> <pre>Input your request (1 book - with area, 2 print, 3 exit): 1 north Dolomiti 120.00 Input your request (1 book - with area, 2 print, 3 exit): 1 west There is no available hotel in the area requested. Input your request (1 book - with area, 2 print, 3 exit): 1 north Fonti 132.00 Input your request (1 book - with area, 2 print, 3 exit): 2 Miramare south 80.00 Gino east 60.00 Imperiale east 200.00 Locanda south 83.00 Siberiano north 140.00 Input your request (1 book - with area, 2 print, 3 exit): 3 <PROGRAM EXITS></pre>
---	--