

Subiecte proba Embedded Systems

Mini stație meteorologică



Scop:

Se cere dezvoltarea unei mini stații meteorologice care să îndeplinească următoarele cerințe.

Echipamente folosite:

- platformă de dezvoltare cu microcontroler ATMEGA
- senzor de temperatură
- senzor de luminozitate
- bec
- led
- PC
- elemente pasive
- tranzistori

Cerințe minime:

- citirea temperaturii – 20 puncte
- citirea informației de luminozitate – 20 puncte
- afisarea datelor pe LCD – 40 puncte
- transmiterea informațiilor către PC doar la un interval fix – 40 puncte
- aplicație PC care să citească și afișeze datele citite – 20 puncte
- aplicația PC trebuie să afișeze și pictograme aferente situației meteo și condiției noapte/zi
20 puncte

Cerințe suplimentare:

- aplicație web să afișeze datele citite de la ministația meteorologică, inclusiv pictogramele online – 20 puncte
- modificarea de pe PC a intervalului de timp la care se actualizează informația – 10 puncte
- modificarea de pe aplicația web a intervalului de timp la care se actualizează informația – 20 puncte
- realizarea citirii “la comandă” a informației meteorologice în aplicația PC – 20 puncte
- realizarea citirii “la comandă” a informației meteorologice în aplicația web – 20 puncte
- realizarea unui grafic de evoluție în timp a datelor achiziționate de la senzor pe PC – 30 puncte
- realizarea unui grafic de evoluție în timp a datelor achiziționate de la senzor pe web – 30 puncte
- conectarea unui bec la stația meteorologică care se aprinde pe timp de noapte – 15 puncte

Bonus:

- modificarea intervalului de reactualizare (+- 1 sec) de la butoanele de pe placă – 10 puncte
- modificarea intervalului de reactualizare cu valoare introdusă de la tastatura de pe placă – 25 puncte

Punctaj total posibil: 360

Notă:

1. Aplicația trebuie dezvoltată astfel încât să poată fi testată de către juriu (exemplu: intervalele de timp de citire să poată fi setate la nivelul secundelor)
2. Condiția care desparte intervalul noapte/zi este dată de ora PC-ului la care e conectată stația meteorologică

Subiecte proba Software Development

Vacation management system



Description

Write an application that manages the vacation time in a company. Company's employees are organized in teams; one person can be member of one or more teams. Each employee can request vacation for a sequence of days or for a time period. There are several types of vacations (e.g. regular vacation, legal holidays, medical, maternal, etc.). There are a maximum number of days available per year for each type, except for holidays, where the number of available days is calculated as: $20 + (\text{years} \% 5)$, meaning the person gets one extra vacation day for every 5 years spent in the company. Vacations can be modified or canceled all together, but only if the requested time interval is in the future. At team level, a configurable percentage of the team members must be present at all times. The application must take into account legal holidays.

Input

Application's input consists of:

- The list of employees, including hiring date.
- The list of teams, including members and percentage of mandatory presence.
- The vacation types, including maximum available days per year, where applicable.
- The list of legal holidays (see additional file for legal holidays in the year 2012).

Requirements

1. Add / modify / cancel vacation requests.
The application must give a meaningful error and stop if the request cannot be processed.
2. Given a name, indicate the number of remaining regular vacation days for the current year.
3. Given a date and a name, return a response indicating if the person will be at work that day, or will be on vacation. In the later case, indicate the type of vacation, and the date of return.
4. Given a time period and a team, create an overview report of the working/vacation slots for each team member
5. Given a time period and a name, return the time slots eligible as vacation.

Important notes

- In the evaluation process we will carefully examine several aspects of your work, not only the functional requirements themselves; the code architecture, user-friendliness and performance, for example, are also of importance. Please check the scoring section for more details.
- We are aware that the subject presented here is challenging. Reaching the maximum score is close to impossible. However, you have plenty of options to maximize your score by adapting to your particular skill-set. Also, we will have a better chance to compare and tell apart different solutions. All applications will be evaluated, however incomplete. And, if we are impressed, your solution may bring you an excellence prize in a specific area (see details in the prizing section).
- You have complete freedom over how you handle inputs and outputs, and this is intentional. Implementing database persistence and/or GUI over file I/O will bring you more points at the cost of extra-work and more time spent. It is for you to balance the pros and cons and decide.
- To be eligible for a prize, an application must cumulate at least 51 points.
- Our recommendation is to have a team size of 2 – 4 members (not more than 4 allowed)
- Every solution must be accompanied by a text file containing:
 - The comprehensive list of tools required to run your application (+ versions, where needed)
 - Installation instructions, where required
 - User guide skeleton (simple description of how to proceed to get the results)

Scoring

The total score is of 151 points, distributed as follows:

- Code architecture – 36 points
(Complete set of classes, modularity, flexibility, testability)
- Functional features (see details below) – 40 points
 - Add / Modify / Delete vacation request (incl. error handling, so on) – 16 points
 - Remaining vacation days – 2 points
 - Check working vs. vacation day – 2 points
 - Team report – 10 points
 - Possible vacation slots – 10 points
- Correctness – 20 points
(Correctly covered test scenarios)
- I/O and presentation – 30 points
(Input data gathering, storage, outputs and/or GUI)
- Performance / scalability – 20 points
(Many large teams with reasonable memory and time consumption)
- Maintainability – 5 points

Prizes

Awards will go to the teams cumulating the highest 3 scores. These scores must be above 51 points. Also, additional prizes may go to excellence in the area of:

- Architecture
- User experience
- Performance and scalability

