|  |  |
| --- | --- |
| **Øvelse** | Power Manager |
| **Projekt** | PowerManager |
| **Formål** | Repetition af flere emner fra 1.semester pensum |
| **Beskrivelse** | Den givne Visual Studio solution rummer de to ufuldstændige klasser **Apparat** og **ElNet**, som tilsammen udgør en simpel model for strøm­forbrug i f.eks. en bolig.  Et **Apparat** repræsenterer en enhed der bruger strøm, f.eks. et TV, en computer, en el-kedel, osv.. Et **Apparat** vil således have et **Navn** og et **StrømForbrug** (målt i Watt).  Et **ElNet** repræsenterer et et-netværk som kan forsyne apparaterne med strøm, hvis man forbinder et apparat med el-nettet. Et **ElNet** vil således have en **Beskrivelse** og en **Kapacitet** (målt i Watt), samt en måde at forbinde et apparat til el-nettet på (i praksis en metode).  Din opgave er at udfylde de to klasse-definitioner, dels så de er i over­ensstemmelse med de beskrivelser der følger nedenfor samt kommen­tar­erne i selv koden, dels så den test-kode der er skrevet i metoden **Kør** i klassen **Test** kan køres (og give de forventede resultater). |
| **Trin** | 1. Download og unzip (VIGTIGT) filen **PowerManager.zip** fra Moodle. 2. Åbn **PowerManager** i Visual Studio. Vi starter i klassen **Apparat**. Klassen rummer fra start kun en **ToString**-metode. Du skal nu implementere to properties **Navn** (af typen ***string***) og **StrømForbrug** (af typen ***int***), samt en passende constructor i klassen, således at der ikke længere er nogen fejl i **ToString**. 3. Gå nu over i klassen **Test**, og ned i metoden **Kør**. Indkommentér (dvs. fjern **//**-symbolet fra starten af linjerne) de kode-linjer der står under **1)**. Det skal være sådan, at det ikke forekommer fejl i disse linjer. Hvis der forekommer fejl, må du gå tilbage til **Apparat**-klassen og rette den til (det kan f.eks. være din definition af constructoren, der skal rettes). 4. Kør programmet. Du skulle gerne få udskrevet information om hver af de fem apparater.   *(opgaven fortsætter på næste side)*   1. Fortsæt nu til klassen **ElNet**. Klasser rummer flere elementer som er klar til brug (et instance field, et par properties, en constructor og **ToString**), mens flere andre elementer ikke er gjort færdige (alle elementer som rummer kommentaren ***TODO – ikke færdig***)***.*** Dan dig et overblik over klassen, så du forstår formålet med de forskellige properties og metoder i klassen (se kom­men­tarerne i koden). Du skal ikke tilføje noget til klassen endnu. 2. Gå tilbage til **Test**, og indkommentér alle de kode-linjer i **Kør**, som er under **2)**, **3)**, **4)**, **5)** og **6)**. Inden du kører programmet, prøv da først at tænke igen­nem, hvad du ville forvente som output fra programmet, hvis ellers alle me­toder og properties virkede som beskrevet. 3. Kør programmet. Forhåbentlig kommer du til den konklusion, at program­met ikke giver det forventede output 😊. Det er – selvfølgelig – fordi vi mangler at færdiggøre implementationen af flere elementer i klassen. 4. Gå tilbage til klassen **ElNet**, og færdiggør implementationen af de ufærdige elementer (det kan være en god ide at prøve at færdiggøre et element ad gangen, køre programmet, og fortsætte på den måde indtil det element ser ud til at virke, og derefter fortsætte til næste element). Det drejer sig om disse elementer:    1. Metoden **Tilslut**    2. Metoden **Udskriv**    3. Den beregnede property **Belastning**    4. Den beregnede property **Overbelastet** 5. Nu har du løst opgaven… men der er flere muligheder for forbedring. Neden­stående liste er nogle forslag du kan overveje. Du bør også sørge for at tilføje kode til **Kør**, som tester de tilføjelser/ændringer du laver.    1. Tilret **Tilslut**-metoden i **ElNet**, så det givne apparat kun bliver tilslut­tet, hvis det ikke medfører at el-nettet så bliver overbelastet.    2. (I forlængelse af a)) I fald det givne apparat ikke bliver tilsluttet, udskriv da en pas­sende besked om dette på skærmen.    3. Gør det muligt for den som opretter et **ElNet**-objekt at angive, om man vil have udskrevet en besked eller ej, hvis et givent apparat ikke bliver tilsluttet.    4. Gør det muligt at tænde/slukke for et apparat, efter det er blevet tilsluttet til el-nettet. Dette skal medføre at logikken for at udregne belastning kun skal medtage de tændte apparater. Det skal således være muligt at tænde/slukke for et apparat ved at kalde en metode på et **ElNet**-objekt, hvor man angiver navnet på apparatet samt om det skal tændes eller slukkes. |