

Herhalingsopdrachten C voor het vak Embedded Systems

Deze opdrachten moeten worden gemaakt en uitgevoerd op een PC. Je mag zelf je ontwikkeltools selecteren (Visual studio, Pelles C, gcc, mingw, ...)

Opdracht 1:

Maak een "hello world" applicatie

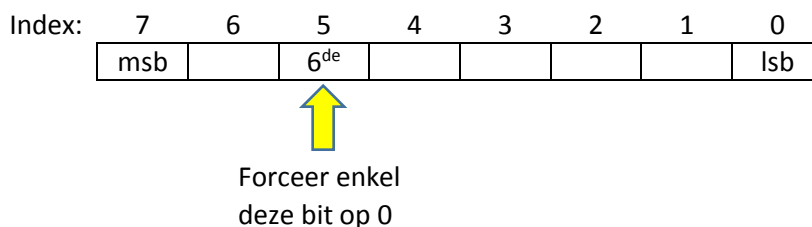
Opdracht 2:

Vraag aan de gebruiker een zin, stockeer deze en print deze af.

Opdracht 3:

In het programma van opdracht 2 gebruik je een for-lus om elk karakter van deze zin te overlopen (enkel de ingegeven karakters en niet de lengte van de buffer).

Gebruik bit operaties om bij elk karakter de 6^{de} bit op 0 te forceren zonder de andere bits aan te passen (6^{de} bit is bit met als index 5). Druk de zin opnieuw af en verklaar het verschil.



Laat uw code aan de docent zien en verklaar de output.

Opdracht 4:

Maak van de logica in opdracht 3 een functie met als naam `to_u`.

Deze functie heeft 1 argument waarmee de string wordt doorgegeven.

De functie past de string zelf aan en hoeft dus ook niets te returnen.

Opdracht 5:

Maak een gelijkaardige functie die het omgekeerde doet van de functie `to_u` en noem deze functie `to_l`.

Opdracht 6:

Begin terug van een leeg programma. Maak een variabele aan met als naam `str` waar de volgende tekst in zit: "Pointers zijn zeer krachtig!\n".

1. Print `str`.
2. Declareer een pointer met als naam `ptr` die naar een karakter kan wijzen.
3. Laat `ptr` naar het begin van `str` wijzen.
4. Tel bij `ptr` 14 bij.
5. Waar de pointer naar wijst schrijf/zet je een "F".
6. Incrementeer de pointer 2 keer.
7. Waar de pointer naar wijst schrijf/zet je een "N".
8. Incrementeer de pointer opnieuw.
9. Waar de pointer naar wijst schrijf/zet je een 0 of een `\0`.
10. Laat de pointer wijzen naar het 16^{de} karakter van `str` (deze heeft index 15).
Gebruik hiervoor `&` en `[]`.

11. Schrijf naar deze plaats een "U".
12. Print str.

Opdracht 7.

Bij deze opdracht is er een bijkomende header "list.h" (zie trac). In deze header is er een gelinkte lijst gedefinieerd.

Deze gelinkte lijst bestaat zoals elke lijst uit nodes. Een node wordt in de code altijd voorgesteld als een structure. Deze structure bestaat uit minimum 2 delen: Data en een pointer naar de volgende node. Sommige gelinkte lijsten hebben nog extra pointers om "speciale" vormen van gelinkte lijsten te maken zoals dubbel gelinkte lijsten, boom structuren, ...

De node in de header bestaat uit een pointer (next) naar de volgende node en een karakter array (data).

De eerste node heet node0.

Maak een lus die over elke node van de gelinkte lijst "loopt" en de data afdruckt.

De pointer in de laatste node is een NULL pointer. Op die manier weet je wanneer de lus moet stoppen.

Normaal kan het in ongeveer 3 lijnen.

