

MASK

We gaan in deze opgave de afzonderlijke bits van een byte manipuleren.

Codeer instructies :

Zie de gegeven code : mask.cpp.

We beperken ons tot het manipuleren van bits in een byte.

De typedef in de voorbeeld code definieert een Byte als een unsigned char (en is dus 1 byte groot).

Evenzo kunnen we UInt gebruiken om een unsigned int aan te duiden.

1. makeMask
parameter: pos, $0 \leq \text{pos} \leq 7$ met 0 = LSB en 7 = MSB.
deze functie moet een resultaat Byte geven die op de opgegeven positie een 1 heeft staan en op alle andere posities een 0.
Merk op dat deze functie altijd een mask geeft met slechts één bit met de waarde 1.
2. setMask
deze functie moet een resultaat Byte geven die op alle posities van source een 1 geven waar mask een 1 heeft. De overige posities van source blijven onveranderd.
Deze functie kan een mask geeft met meer dan één bit met de waarde 1.
3. unsetMask
deze functie moet een resultaat Byte geven die op alle posities van source een 0 geven waar mask een 1 heeft. De overige posities van source blijven onveranderd.
4. checkMask
deze functie geeft een waarde **true** als de source een 1 heeft op alle posities waar de mask een 1 heeft.

Hints :

Gebruik hier de binaire operaties : **~**, **&** en **|** .

Conditie:

Bij makeMask moet een controle op de waarde van de parameter plaatsvinden.

MASK

Gegeven is de volgende code :

mask.cpp:

```
typedef unsigned char Byte;
typedef unsigned int Uint;

Byte makeMask(Uint pos){
    // code goes here ...
    return ... ;
}

Byte setMask( Byte source, Byte mask ){
    // code goes here ...
    return ... ;
}

Byte unsetMask( Byte source, Byte mask ){
    // code goes here ...
    return ... ;
}

bool checkMask(Byte source, Byte mask){
    // code goes here ...
    return ... ;
}

int main()
{
    // todo ...
    return 0;
}
```

Merk op:

1. De inhoud van de '**main**' routine is niet van belang voor de test.
Je mag hier test-code opnemen voor de gemaakte functies, geheel naar eigen inzicht.
2. De 'signatuur' van de functies mag niet worden aangepast. Wijzigingen geven fouten in de test.

Upload instructies:

Upload de volgende bestanden:

1. mask.cpp