

Grafer – Case 2:

(Basert på oppg.nr.3 på eksamen 12/12-1995 i LO164A-Algoritmiske metoder)

Vi har følgende const'er, struct og globale data:

```
const int MAXNODER = 100;    ///< Max.antall noder.
const int MAXNABOER = 10;    ///< Max.antall naboer for hver node.

struct Node {
    int ID;                    // Nodens ID/navn/key.
    int antKanter;            // Antall naboer/kanter ut fra node.
    Node* nabo[MAXNABOER];    // Pekere til naboene.
    Node* kopi;                // Peker til evt. duplikat av en selv.
    Node(const int id, const int ant) {
        ID = id; antKanter = ant; kopi = nullptr; }
};

Node* gNoder[MAXNODER];      ///< Array med peker til hver enkelt node i
                             ///< grafen. Brukes kun i "kopierIterativt".
                             ///< Bruker indeksene 0 til gTotAntNoder-1
int gTotAntNoder;            ///< Totalt antall noder i grafen.
Node* gGrafen;                ///< Peker til startnoden i hele grafen.
```

I hele denne oppgaven skal vi se på *to ulike funksjoner* som hjelper oss med å lage en *kopi* av en graf. Dvs. den opprinnelige grafens *kopi* skal etterpå peke til en node som er en kopi av den selv, der ID og antKanter er identiske, *nabo[i]* i kopien peker til *neste[k]* sin kopi. Pekeren *kopi* i kopien skal fortsatt være *nullptr*.

Opprettelsen av grafen i hukommelsen, og initieringen av *gGrafen* skjer «automatisk».

a) Lag den ikke-rekursive funksjonen void kopierIterativt()

som lager en kopi av grafen som beskrevet ovenfor. Funksjonen bør gå gjennom *gNoder* to ganger. Først en gang for å opprette en kopi av hver enkelt node. Og deretter en gang til for å opprette pekere mellom kopinodene, tilsvarende de i den originale grafen.

b) Nå er hverken *gNoder* eller *gTotAntNoder* lengre tilgjengelig. I stedet kan vi nå *hele* grafen via *gGrafen*.

Lag den rekursive funksjonen void kopierRekursivt(Node* node)

som også kopierer hele grafen som beskrevet ovenfor.

Den startes fra main ved kallet: kopierRekursivt(gGrafen)

Hint: Bruk pekeren *kopi* til å sjekke om en node er SETT/USETT