

**Minitoets 2.5 Propedeuse Electrical Engineering**  
**Maandag 13-12-2010; 9.45 – 10.30 uur**  
**Onderdeel Programmeren in C (EE1400) (± 22 minuten)**

Naam:

Cijfer:

Studienummer:

*Bij deze toets mag je een boek over C programmeren, de C practicumhandleiding en prints van de collegeslides erbij houden. De puntenverdeling is als volgt: vraag 1 (1.5), 2 (1.5), 3(3) en 4(3), totaal 9 punten*

---

1. Gegeven een pointer `ptr` naar de volgende structure:

```
struct date {
    int year;
    int month;
    int day;
};
```

Geef twee verschillende expressies in C om aan member `month` te refereren:

- a. `ptr -> month`
- b. `(*ptr).month`

Verder gegeven de code

```
struct person {
    char name[32];          /* naam */
    struct date born;      /* geboortedatum */
    struct date died;      /* overlijdensdatum */
};
struct person king1;
```

Geef de expressies in C om aan het geboortjaar van `king1` te refereren:

`king1.born.year`

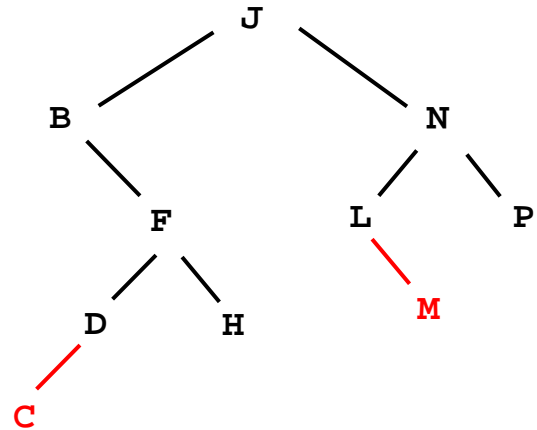
2. Wat zal worden afgedrukt door het volgende stukje C code ?

```
char strA[] = "10 20 dertig";
int a, b, c, d;
a = b = c = 40;
d = sscanf (strA, "%d%d%d", &a, &b, &c);
printf ("%d %d %d %d\n", d, c, b, a);
```

`2 40 20 10`

4. Gegeven de nevenstaande gesorteerde binary tree. Geef in de tree duidelijk aan (gaat de verbindende lijn schuin naar links of schuin naar rechts ?) waar de nodes met de letters C en M geplaatst zouden worden.

Member left van een node verwijst naar de node linksonder en member right verwijst naar de node rechtsonder van deze node. Wanneer de waarde van zo'n member NULL is, is de desbetreffende node niet aanwezig. Wat zal worden geprint bij het aanroepen van de volgende functie met node B (!) als argument (laat C en M buiten beschouwing) ?



```

void visit (struct node *n)
  if (n != NULL) {
    visit (n -> left);
    visit (n -> right);
    printf ("%c ", n -> data);
  }
}

```

D H F B

5. Gegeven de volgende structure voor een lineaire linked list:

```

struct list {
  int      data;
  struct list * next;
};
typedef struct list List;

```

Schrijft een functie listlength waarbij een pointer naar het begin van de list als argument wordt meegegeven, en waarbij het aantal elementen in de list als integer wordt getouneerd.

```

int listlength (List *lst)
{
  int cnt = 0;

  while (lst != NULL) {
    lst = lst -> next;
    cnt++;
  }
  return (cnt);
}

```