

Tentamen Programmeren in C (EE1400)

3 februari 2012, 9.00 – 12.00

Uitwerkingen

Opgave 1 (24 punten)

- a. 6 11
- b. 8 16
- c. 12 3
- d. 34 32
- e. 17 76
- f. n 1

Opgave 2 (8 punten)

- a. -9
- b. 13
- c. 81
- d. 61

Opgave 3 (10 punten)

```
int sum, end, i;
char s[128];

sum = 0;
end = 0;
scanf ("%s", s);
i = 0;
while (!end) {
    switch (s[i]) {
        case '1':
            sum += 1;
            break;
        case '2':
            sum += 2;
            break;
        default:
            end = 1;
            break;
    }
    i++;
}
printf ("sum=%d\n", sum);
```

Opgave 4 (6 punten)

De functie telt telkens een waarde 1 af van de beginwaarde, net zolang kleiner dan 1 is. Het aantal malen dat er afgetrokken, zo veel maal wordt er vermenigvuldigd met 2, waarbij de beginwaarde 4 is.

```
int func (int x)
{
    int a = 4;

    while (--x >= 1) {
        a = 2 * a;
    }

    return a;
}
```

Opgave 5 (8 punten)

Drie verschillende mogelijke oplossingen zijn:

```
int str2int (char *s)
{
    int i, v = 0, d = 1;

    for (i = strlen (s) - 1; i >= 0; i--) {
        v += d * (s[i] - '0');
        d *= 10;
    }
    return v;
}
```

```
int str2int (char *s)
{
    int v = 0;

    while (*s) {
        v = v * 10 + (*s++ - '0');
    }
    return v;
}
```

```
int str2int (char *s)
{
    int v;
    sscanf (s, "%d", &v);
    return v;
}
```

Opgave 6 (8 punten)

```
double variance (double x[], int n, double u)
{
    int i;
    double val = 0.0;

    for (i = 1; i <= n ; i++) {
        val += (x[i] - u)*(x[i] - u);
    }
    return val/n;
}
```

Opgave 7 (8 punten)

```
int tel_namen (struct burger *lijst, char *naam)
{
    int aantal = 0;
    while (lijst) {
        if (strcmp (lijst -> voornaam, naam) == 0) {
            aantal++;
        }
        lijst = lijst -> volgend;
    }
    return (aantal);
}
```

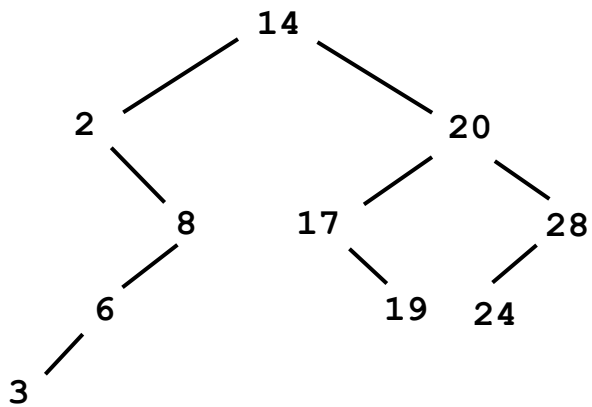
Opgave 8 (8 punten)

```
void calculate_totals(int a[M][N], int b[N])
{
    int i, j, grand_total = 0;

    for (j = 0 ; j < N; j++) {
        b[j] = 0;
        for (i = 0 ; i < M ; i++) {
            b[j] += a[i][j];
        }
        grand_total += b[j];
    }
    return (grand_total);
}
```

Opgave 9 (10 punten)

a.



b.

```
int tree_sum (struct node * n)
{
    if (n) {
        return (n -> data
                + tree_sum (n -> left)
                + tree_sum (n -> right));
    }
    else
        return 0;
}
```