

In this document, you will find a few C program listings rendered with the color scheme used in the Code::Blocks IDE.

```
1  #include <stdio.h>
2  #include <math.h>
3
4  int main(void) {
5      double x, y, wortel, macht;
6      int i;
7      printf("Voer een positief getal x in: ");
8      scanf("%lf", &x);
9      printf("Voer een getal y in: ");
10     scanf("%lf", &y);
11     for (i = 0; i < 5; i = i + 1) {
12         wortel = sqrt(x);
13         macht = pow(x, y);
14         printf("De vierkantswortel uit %f is: %f\n", x, wortel)
15             ↪ ;
16         printf("%f tot de macht %f is: %f\n", x, y, macht);
17         x = x + 1;
18     }
19     fflush(stdin);
20     getchar();
21     return 0;
22 }
```

```
1  #include <stdio.h>
2
3  int main(void){
4      int cijfer;
5      char letter;
6      do {
7          printf("Geef je cijfer: ");
8          scanf("%d", &cijfer);
9      } while (cijfer < 0 || cijfer > 10);
10     switch (cijfer) {
11         case 10:
12         case 9:
13         case 8:
14             letter = 'A'; break;
15         case 7:
16             letter = 'B'; break;
17         case 6:
18             letter = 'C'; break;
19         case 5:
20             letter = 'D'; break;
21         default:
22             letter = 'F'; break;
23     }
24     printf("Dit komt overeen met een %c.\n", letter);
25     fflush(stdin);
26     getchar();
27     return 0;
28 }
```

```
1  #include <stdio.h>
2
3  int fib(int n) {
4      if (n < 2) {
5          return n;
6      }
7      else {
8          return fib(n-1) + fib(n-2);
9      }
10 }
11
12 int main(void) {
13     int getal;
14
15     do {
16         printf("Geef een getal groter dan of gelijk aan 0: ");
17         scanf("%d", &getal);
18     } while (getal < 0);
19
20     printf("fib(%d) is %d\n", getal, fib(getal));
21
22     printf("Druk op enter om het programma af te sluiten.");
23     fflush(stdin);
24     getchar();
25     return 0;
26 }
```

```
1  #include <stdio.h>
2
3  /* Dit programma demonstreert hoe een
4     groot getal cijfer voor cijfer kan
5     worden ingelezen */
6
7  int leesCijfer(void) {
8      switch (getchar()) {
9          case '0': return 0;
10         case '1': return 1;
11         case '2': return 2;
12         case '3': return 3;
13         case '4': return 4;
14         case '5': return 5;
15         case '6': return 6;
16         case '7': return 7;
17         case '8': return 8;
18         case '9': return 9;
19         default: return -1;
20     }
21 }
22
23 int main(void) {
24     int cijfer;
25     printf("Type een groot getal:\n");
26     do {
27         cijfer = leesCijfer();
28         if (cijfer != -1) {
29             printf("%d ", cijfer);
30         }
31     } while (cijfer != -1);
32
33     return 0;
34 }
```

```
1  #include <stdio.h>
2
3  #define AANTAL 10
4
5  int main(void) {
6      double temperatuur[AANTAL];
7      int i;
8      double temp_acc;
9
10     for (i = 0; i < AANTAL; i = i + 1) {
11         do {
12             printf("Geef temperatuur %d op: ", i + 1);
13             fflush(stdin);
14             } while (scanf("%lf", &temperatuur[i]) != 1);
15     }
16
17     temp_acc = 0.0;
18     for (i = 0; i < AANTAL; i = i + 1) {
19         temp_acc = temp_acc + temperatuur[i];
20     }
21
22     printf("De gemiddelde temperatuur is %.2f\n\n", temp_acc /
23           ↪ AANTAL);
24
25     printf("Druk op enter om het programma af te sluiten.");
26     fflush(stdin);
27     getchar();
28     return 0;
29 }
```

```
1 #include <stdio.h>
2 #include <malloc.h>
3
4 int main(void) {
5
6     char *ptr = malloc(1024); /* Allocate 1024 bytes */
7
8     if (ptr == NULL) {
9         printf("Oops, no memory!\n");
10        exit(-2);
11    }
12
13    printf("Allocated 1024 bytes at address %p\n", ptr);
14
15    free(ptr); /* Free memory */
16
17    return 0;
18 }
```

```
1 char *ptrnew;
2
3 ptrnew = realloc(ptr, 2048);
```

```
1  /* myprog.c */
2  #include <stdio.h>
3  #include <stdlib.h>
4
5  int main(int argc, char *argv[]) {
6
7      int i;
8
9      printf("\nAantal argumenten: %d\n\n", argc);
10     for (i = 0; i<argc; i++) {
11         printf("Argument %d: %s\n", i, argv[i]);
12     }
13     return 0;
14 }
```
