

Kurseinheit X: Ergänzungen

1. Interner Buffer scanf

1. Interner Buffer scanf

Keyboard-Buffer und scanf-Buffer

Keyboard-Buffer:

Eingaben über die Tastatur liegen im Keyboard-Buffer! Mit Funktionen wie `_kbhit()` kann dieser abgefragt werden.

Sobald `scanf` aufgerufen wird dieser Inhalt in einen internen „statischen“ Buffer von `scanf` übernommen. Auch alle Eingaben, die die Funktion einliest, landet in diesem Buffer.

Kann dieser Buffer nicht in das gewünschte Format konvertiert werden (z.B. in ein `double`), so verbleiben die Eingaben weiterhin in diesem Buffer und werden beim nächsten `scanf`-Aufruf verwendet. Daher gibt es bei Fehleingaben diese Effekte, dass Eingaben übersprungen bei `scanf` werden. Daher müsste nach jedem `scanf` (außer bei `%s`) die folgende Zeile stehen.

```
int c;  
while( (c = getchar()) != '\n' && c != EOF);
```

<https://itqna.net/questions/14240/clear-c-buffer-fflush-or-fpurge>

1. Interner Buffer scanf

3

Beispiel

```
int main(void)
{
    int c;
    char acText[50];
    float f;
    double d;

    printf("Eingabe f:\n");
    scanf("%f", &f);
    while ((c = getchar()) != '\n' && c != EOF);

    printf("Eingabe d:\n");
    scanf("%lf", &d);
    while ((c = getchar()) != '\n' && c != EOF);

    printf("Eingabe text:\n");
    scanf("%s", &acText);

    return 0;
}
```

„Upps-Something“

Ohne while-Schleife:

„Upps-Something“ kann nicht in float konvertiert werden. Bleibt im Buffer.

„Upps-Something“ kann nicht in double konvertiert werden. Bleibt im Buffer.

„Upps-Something“ wird einlesen. Eigene Eingabe ist dann zu spät.