

Vad förväntas ni kunna

Repetition

- Läsa från tangentbordet (standard in)
 - `getc` – läser ett tecken från angiven inström (vanligtvis fil eller `stdin`)
 - `getchar` – läser ett tecken från tangentbordet (`stdin`).
 - `putchar` – skriver ut ett tecken på skärmen (`stdout`)
 - `putc` – skriver ut ett tecken till angiven utström (fil eller `stdout`)
 - Sök slutet på filen.
- Upprepning
 - `while`
 - `for`

getc/getchar/putc/putchar – Skillnader och likheter.

getc – Kan läsa från tangentbordet (stdin) och från filer.

getchar – Kan läsa ett tecken från tangentbordet (stdin).

putc – Skriver ut ett tecken till angiven utström (fil eller stdout)

putchar – Skriver ut ett tecken på skärmen (stdout)

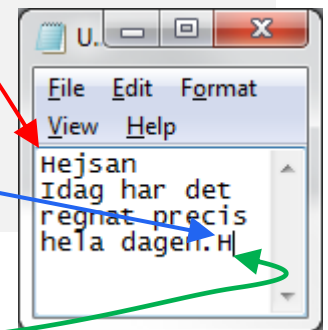
getc och putc – kodexempel.

Vi läser från standard in (tangentbordet).

```
#include <stdio.h> //Innehåller stdin, stdout, getc, putc
int main() {
    char c;
    printf("Enter character: ");
    c = getc(stdin); // Vi läser ett tecken från tangentbordet med getc
    printf("Character entered: ");
    putc(c, stdout); // Skriver ut ett tecken på skärmen med putc
    system("PAUSE");
    return (0);
}
```

Vi läser från textfilen dagbok.txt

```
#include <stdio.h> // Innehåller FILE
int main() {
    char c;
    FILE *fil = fopen ("dagbok.txt", "a+"); // Vi öppnar filen för läsning och skrivning
    c = getc(fil); // Läser ett tecken från början av filen
    fseek(fil, 0, SEEK_END); // Vi hoppar till slutet av filen
    putc(c, fil); // Skriver ut ett tecken till filen
    fclose(fil);
    system("PAUSE");
    return (0);
}
```



Läromål: Kunna flytta filpekaren till slutet av
filen. Ha en ökad förståelse för getc och putc.

getchar/putchar – kodexempel.

Vi läser från standard in (tangentbordet). getc och putc.

```
#include <stdio.h>
int main() {
    char c;
    printf("Enter character: ");
    c = getc(stdin);
    printf("Character entered: ");
    putc(c, stdout);
    system("PAUSE");
    return (0);
}
```

Vi läser från textfilen dagbok.txt

```
#include <stdio.h> // Innehåller FILE
int main() {
    char c;
    FILE *fil = fopen ("dagbok.txt", "a+");
    c = getc(fil);
    fseek(fil, 0, SEEK_END);
    putc(c, fil);
    system("PAUSE");
    return (0);
}
```

```
#include <stdio.h> //Innehåller getchar, putchar

int main() {
    char c;
    printf("Enter character: ");
    c = getchar(); // Vi läser ett tecken från standard in (tangentbordet) med getchar
    printf("Character entered: ");
    putchar(c); // Skriver ut ett tecken till standard ut (skärmen) med putchar.
    system("PAUSE");
    return (0);
}
```

for och while

- Ganska lika
 - Man kan ofta använda vilken som helst av dem
 - Några skillnader:
 - for används när man har
 - En lista på saker som man ska göras samma sak med.
 - Ett bestämt antal gånger som något ska göras.
 - Användaren är inte tänkt att kunna bestämma när for-satsen ska avslutas
 - Om man inte ska använda uppräknaren efter for-satsen.
- Ex.: `for(uppräknare=0; uppräknare<MAX, uppräknare++){`

while används när man ska göra något

- Så länge ett visst villkor är sant.
- Ett okänt antal gånger som något ska göras
- Användaren är tänkt att bestämma när while-satsen ska avbrytas.
- Om man tänker använda uppräknaren efter while-satsen. Ex.:


```
int uppräknare=0;
while (uppräknare<MAX) {
    uppräknare++;
}
printf("%i", uppräknare);
```

Notera att man kan inte använda bokstaven ä i variabelnamn!

for

```
#define ANTAL 100
int main(void){
    int gjordaForVarv, antalStora = 0, slumpTalet = -1;
    srand(time(0));
    for(gjordaForVarv = 0; gjordaForVarv < ANTAL; gjordaForVarv++) {
        slumpTalet = rand()%100 + 1;
        printf("%i\n", slumpTalet);
        if (slumpTalet > (ANTAL / 2)) antalStora++;
    }
    printf("Antalet slumpstal större än %i blev: %i\n", (ANTAL/2), antalStora);
    system("PAUSE");
    return(0);
}
```

while

```
#define ANTAL 100
int main(void){
    int gjordaForVarv = 0, antalStora = 0, slumpTalet = -1;
    srand(time(0));
    while (gjordaForVarv < ANTAL) {
        slumpTalet = rand()%100 + 1;
        printf("%i\n", slumpTalet);
        if (slumpTalet > (ANTAL / 2)) antalStora++;
        gjordaForVarv++;
    }
    printf("Antalet slumpstal större än %i blev: %i\n", (ANTAL/2), antalStora);
    system("PAUSE");
    return(0);
}
```

while

```
int main(void) {
    char namn[] = "Slagersm";

    putchar(namn[0]);
    putchar(namn[1]);
    putchar(namn[2]);
    putchar(namn[3]);
    putchar(namn[4]);
    putchar(namn[5]);
    putchar(namn[6]);
    putchar(namn[7]);

    puts("");
    system("PAUSE");
    return(0);
}
```

```
int main(void) {
    int n = 0;
    char namn[] = "Slagersm";

    while (n<8) {
        putchar(namn[n++]);
    }

    puts("");
    system("PAUSE");
    return(0);
}
```

Vad förväntas ni kunna

Repetition

- Läsa från tangentbordet (standard in)
 - `getc` – läser ett tecken från angiven inström (vanligtvis fil eller `stdin`)
 - `getchar` – läser ett tecken från tangentbordet (`stdin`).
 - `putchar` – skriver ut ett tecken på skärmen (`stdout`)
 - `putc` – skriver ut ett tecken till angiven utström (fil eller `stdout`)
 - Sök slutet på filen.
- Uppprepning
 - `while`
 - `for`